

TRABALHO DE CONCLUSÃO DE CURSO

ESTIMAÇÃO DE DIREÇÃO DE CHEGADA DE  
SINAL SONORO UTILIZANDO ARRANJO DE  
SENSORES

Autor: Guilherme de Sousa Torres  
Orientador: Prof. Ricardo Zelenovsky

Brasília, junho de 2018

**UNIVERSIDADE DE BRASÍLIA**

FACULDADE DE TECNOLOGIA

**UNIVERSIDADE DE BRASÍLIA**  
**FACULDADE DE TECNOLOGIA**  
**DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**ESTIMAÇÃO DE DIREÇÃO DE CHEGADA DE SINAL SONORO**  
**UTILIZANDO ARRANJO DE SENSORES**

**GUILHERME DE SOUSA TORRES**

**ORIENTADOR: RICARDO ZELENOVSKY**

**TRABALHO DE CONCLUSÃO DE CURSO EM ENGENHARIA**  
**ELÉTRICA**

**BRASÍLIA/DF, JUNHO - 2018**

**UNIVERSIDADE DE BRASÍLIA**  
**FACULDADE DE TECNOLOGIA**  
**DEPARTAMENTO DE ENGENHARIA ELÉTRICA**


**ESTIMAÇÃO DE DIREÇÃO DE CHEGADA DE SINAL SONORO**  
**UTILIZANDO ARRANJO DE SENSORES**


GUILHERME DE SOUSA TORRES

TRABALHO DE CONCLUSÃO DE CURSO SUBMETIDO AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA OBTENÇÃO DO GRAU DE ENGENHEIRO ELETRICISTA.

APROVADA POR:

  
\_\_\_\_\_  
Prof. Dr.: Ricardo Zelenovsky, UnB/ENE  
Orientador

  
\_\_\_\_\_  
Prof. Dr.: Alexandre Ricardo Romariz, UnB/ENE  
Professor Convidado

  
\_\_\_\_\_  
Prof. Dr.: Daniel Chaves Café, UnB/ENE  
Professor Convidado

BRASÍLIA, 28 DE JUNHO DE 2018

## **FICHA CATALOGRÁFICA**

**TORRES, GUILHERME DE SOUSA**

Estimação de Direção de Chegada de Sinais Sonoros Utilizando Arranjo de Sensores [Distrito Federal], 2018.

Trabalho de Conclusão de Curso – Universidade de Brasília, Faculdade de Tecnologia  
Departamento de Engenharia Elétrica. Orientação: Prof. Dr. Ricardo Zelenovsky.

1. Estimação DOA
2. Arranjo linear de sensores
3. Plataforma Embarcada

## **REFERÊNCIA BIBLIOGRÁFICA**

TORRES, GUILHERME DE SOUSA (2018). Estimação de Direção de Chegada de Sinais Sonoros Utilizando Arranjo de Sensores. Trabalho de Conclusão de Curso em Engenharia Elétrica, 2018, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 51p.

*Aos meus Entes queridos.*

## AGRADECIMENTOS

*Agradeço a Deus pela infinidade de oportunidades que Ele me concedeu, capacitando-me e guiando-me sempre, onde quer que eu esteja.*

*À minha família pelo apoio incondicional durante toda a minha formação. Agradeço, em especial, à minha mãe, minha avó Joana, meu tio Ricardo e meu pai, por serem os maiores exemplos de amor e perseverança que um homem poderia encontrar.*

*Ao meu orientador, Prof. Ricardo Zelenovsky, por toda a paciência e dedicação durante todos os momentos.*

*Aos professores Daniel Café e Alexandre Romariz, por aceitarem o convite para compor a banca de avaliação.*

*Aos meus amigos e colegas de curso, por serem uma fonte inesgotável de ânimo e de bons momentos. Juntos, passamos por uma infinidade de situações complicadas, mas graças a vocês consegui manter-me firme em minha caminhada. Obrigado pelas risadas e pela paciência, em especial aos amigos: Pedro, Guilherme, Warley, Jonas, Marcos, Daniel, Augusto, Túlio, Lukas, Kássia, Jéssica, Lúcio.*

*À Giovanna, que jamais deixou de acreditar em mim. Obrigado pelo carinho e compreensão, sempre me motivando a ser melhor, e por ser minha fonte de esperança nos momentos mais difíceis. Obrigado, especialmente, por toda sua paciência inabalável ao ouvir meus incontáveis monólogos sobre a vida, o universo e tudo mais.*

## RESUMO

Este trabalho aborda o estudo e aplicação de um novo método, de baixo custo computacional, para a estimação da direção de chegada – do inglês, DOA (*Direction of Arrival*) – a partir de um sinal sonoro emitido em ambiente fechado.

Para captar os sinais, utiliza-se um arranjo linear de dois microfones, construídos por alunos da Universidade de Brasília. O método investigado consiste na medida de similaridade entre os sinais captados, fazendo medidas de erro quadrático médio entre as amostras para determinar o atraso em tempo discreto.

A fim de avaliar a eficiência do método proposto, são executados ensaios em ambientes estáticos controlados, realizando variados testes de hardware e software para extrair a direção de chegada da maneira mais eficiente possível.

## ABSTRACT

*The present work handles the study and application of a new method of DOA (Direction of Arrival) estimation, from a sound signal sent in a controlled environment.*

*To capture the signals, a linear array of two microphones – both previously built by students of Universidade de Brasília – is utilized. The researched method consists of similarity measurement between both acquired signals. The measurement is done through the calculation of the root mean squared error of the samples to define the discrete time delay.*

*To evaluate the efficiency of the proposed method, various tests are executed inside of static environments, testing both hardware and software to extract the direction of arrival in the most efficient way.*



## SUMÁRIO

1	AMBIENTAÇÃO .....	15
1.1	Motivação:.....	15
1.2	Organização:.....	16
2	EMBASAMENTO .....	18
2.1	Arranjo de sensores .....	18
2.2	Ambiguidade e distância entre sensores .....	21
2.3	Bandas de frequência.....	21
2.4	Filtros digitais .....	23
3	HARDWARE .....	25
3.1	Microfone e circuito pré-amplificador:.....	25
3.2	Arduino MEGA 2560.....	28
3.3	Memória externa.....	30
4	MÉTODO DE ESTIMAÇÃO DE DOA .....	31
4.1	Abordagem em tempo discreto .....	32
4.2	Primeira etapa: Identificação da faixa de maior energia .....	34
4.3	Segunda etapa: Aplicação de um filtro de média móvel .....	35
4.4	Terceira etapa: Interpolação por spline cúbica.....	35
4.5	Quarta etapa: Estimação da direção de chegada .....	35
4.6	Norma euclidiana .....	36
4.7	Defasagem em tempo discreto .....	36
4.8	Múltiplas estimativas .....	38
4.9	Resolução do estimador e limitações .....	39
5	RESULTADOS .....	41
5.1	Avaliação de resultados .....	47
5.2	Limitações do projeto.....	48

6	CONCLUSÃO .....	50
	REFERÊNCIAS BIBLIOGRÁFICAS .....	51

## **LISTA DE TABELAS**

Tabela 1: Médias das 15 aquisições de DOA para cada ângulo de referência.....	47
Tabela 2: Frequência principal do assovio detectada em diferentes situações .....	47

## LISTA DE FIGURAS

Figura 2.1: Sinal $s(t)$ incidindo sobre arranjo linear uniforme de $N$ microfones.....	18
Figura 2.2: Geometria para uma frente de onda plana .....	19
Figura 2.3: Perfil espectral de uma voz masculina .....	22
Figura 2.4: Perfil espectral de um assovio.....	23
Figura 2.5: Design de filtro IIR em ambiente MATLAB.....	24
Figura 3.1: Microfone acoplado à placa do circuito utilizado .....	25
Figura 3.2: Resposta de magnitude versus frequência do microfone de eletreto .....	26
Figura 3.3: Circuito do estágio de pré-amplificação.....	26
Figura 3.4: Alimentação do circuito .....	28
Figura 3.5: Placa Arduino MEGA 2560 utilizada no projeto.....	28
Figura 3.6: Conexão Arduino-SRAM.....	30
Figura 3.7: Arranjo montado.....	31
Figura 4.1: Montagem proposta de arranjo com dois microfones .....	31
Figura 4.2: Janela completa de sinal sonoro captado por um microfone.....	34
Figura 4.3: Representação do deslocamento em tempo discreto efetuado entre dois sinais de tempo discreto.....	37
Figura 4.4: Janela de erro quadrático médio calculado entre sinais sonoros captados pelos microfones .....	38
Figura 4.5: Fluxograma que resume o método de estimação de DOA .....	41
Figura 5.1: Sinais $X_1$ e $X_2$ antes e depois da filtragem.....	42
Figura 5.2: Demonstração da detecção da janela de maior energia.....	42
Figura 5.3: Forma de onda ao longo do tempo onde se observa a defasagem entre os sinais $X_1$ e $X_2$ em uma aquisição .....	43
Figura 5.4: Resultado do erro quadrático médio entre $X_1$ e $X_2$ .....	43
Figura 5.5: Azimute detectado para um ângulo de cerca de $20^\circ$ .....	44
Figura 5.6: Histograma de estimativas para sinais com $\theta = 0^\circ$ .....	45
Figura 5.7: Histograma de estimativas para sinais com $\theta = 30^\circ$ .....	45
Figura 5.8: Histograma de estimativas para sinais com $\theta = 40^\circ$ .....	46
Figura 5.9: Histograma de estimativas para sinais com $\theta = -30^\circ$ .....	46
Figura 5.10: Demonstração de como o mesmo ângulo pode valer para duas fontes diferentes.....	49

## LISTA DE SÍMBOLOS

### Símbolos Gregos

$\theta$	Azimute de incidência	[rad]
$\phi$	Ângulo de elevação	[rad]
$\varphi$	Defasagem angular	[rad]
$\lambda$	Comprimento de onda	[m]

### Outros Símbolos

$d$	Distância entre elementos em um arranjo de sensores	[m]
$v$	Velocidade da onda	[m/s]
$f$	Frequência da onda	[Hz]
$t$	Tempo	[s]
$f_{ams}$	Frequência de amostragem do sinal	[Hz]
$\tau$	Atraso entre sinais de tempo contínuo	[s]
$k$	Atraso em tempo discreto	
$a(\theta_N)$	Vetor volante, ou <i>steering vector</i>	
$\vec{x}$	Vetor de amostras em tempo discreto	
$\vec{e}$	Vetor de erro quadrático médio	

## **Siglas**

ADC	<i>Analogic to digital converter</i>
A/D	Analógico para digital
DOA	Direction of arrival
FFT	<i>Fast Fourier Transform</i>
MATLAB <sup>®</sup>	Marca registrada da MathWorks, Inc.
SNR	<i>Signal – Noise Ratio</i>
SPI	Serial Peripheral Interface
ULA	Uniform Linear Array
DC	<i>Direct Current</i>

# 1 AMBIENTAÇÃO

## 1.1 Motivação:

A capacidade de localizar fontes sonoras é uma habilidade extremamente útil que muitos seres vivos possuem, permitindo a estes uma maior percepção do espaço e dando-lhes a capacidade de reagir ao mundo em sua volta. Certos animais, como morcegos e golfinhos, utilizam-se de ecolocalização para detectar posição e/ou distância de objetos no ambiente. O próprio ser humano possui a capacidade de detectar a posição de sinais sonoros, embora nossa habilidade seja consideravelmente inferior à de tantos outros animais. Entretanto, a observação da natureza e os avanços dos conhecimentos científicos possibilitaram-nos, eventualmente, criar diferentes sistemas de localização sonora, dando às máquinas a capacidade de “ouvir” como nós ouvimos e extrair informações úteis a partir disto.

Para processar um sinal analógico como o som, é necessária a utilização eficiente de um conjunto de sensores usualmente conectados a conversores analógico-digital. Estes sensores permitem-nos captar, guardar e processar sinais sonoros, e há vários anos tem-se obtido consideráveis avanços nestes três aspectos. A filtragem de ruído e limitação em banda, por exemplo, permitem a obtenção de sinais com maior qualidade e fidelidade à fonte. A digitalização do sinal, por sua vez, permite-nos registrar o sinal em memória para utilização posterior. Já a eficiência do processamento do sinal digital está intimamente ligada à capacidade dos computadores e aos algoritmos implementados.

Na área de Controle e Automação, por exemplo, é de suma importância que um robô possa se localizar corretamente, perceber o mundo à sua volta e encontrar pontos de interesse, e não faltam propostas de sistemas sensoriais aplicados a robôs, utilizando câmeras, microfones, GPS, entre outros sensores. Cada sistema possui suas particularidades, e podem se complementar de maneira eficaz a fim de cobrir as limitações uns dos outros [3].

Este trabalho foca na localização de uma fonte sonora presente em um ambiente fechado, com o objetivo de desenvolver um sistema embarcado de baixo custo que seja capaz de estimar a direção de chegada de um sinal sonoro. Na literatura científica, há variados algoritmos de estimação da direção de chegada de um sinal, como CAPON, MUSIC, DS, ESPIRIT e SPR-PHAT [2]. Neste trabalho, contudo, buscou-se alcançar tal objetivo por meio do desenvolvimento de algoritmos que exijam o mínimo de esforço computacional, adequando a parte de processamento a um microcontrolador acessível. O trabalho foi realizando tendo em mente a utilização do sistema embarcado em uma plataforma móvel que, após obter o sinal de

referência a partir da estimaco confivel da direco de chegada do som, altera sua posio e se desloca em direco  fonte sonora.

A fim de obter o resultado pretendido, o projeto descrito neste documento foi dividido em vrias etapas. Durante os primeiros meses, foi feita uma extensa reviso bibliogrfica da literatura que trata sobre estimaco de DOA. O autor baseou-se inicialmente em trabalhos j realizados [1, 2, 6, 7, 8] a fim de investigar diferentes mtodos utilizados na estimaco. Em seguida, construiu-se um circuito pr-amplificador mais simples para estudar a etapa de pr-amplificao.

No decorrer do projeto, foi realizada uma reviso de hardware e software, buscando meios de se realizar a converso A/D dos sinais de interesse da maneira mais eficiente possvel. Os circuitos previamente montados foram abandonados devido ao elevado rudo que adicionavam ao sinal de entrada, sendo substituídos por circuitos implementados anteriormente por alunos da Universidade de Braslia, fornecidos pelo orientador do projeto.

O estgio final consistiu na efetuao de diversos ensaios estticos com o hardware e o software propostos, realizando simulaes em ambiente MATLAB a fim de verificar a eficincia das etapas de digitalizao, processamento e estimaco da direco de chegada dos sinais.

## **1.2 Organizao:**

O Captulo 2 versa sobre todo o embasamento terico por trs deste trabalho, abordando o arranjo linear de microfones e os conceitos-chave que sero utilizados em captulos posteriores.

O Captulo 3 trata sobre o Hardware utilizado, apresentando uma breve explicao acerca dos materiais utilizados neste projeto, como sensores, processador e memria externa.

No Captulo 4  detalhado o mtodo de estimaco de direco de chegada – do ingls DOA (*Direction of Arrival*).  dividido em vrias sees, que abordam as vrias etapas do processo, desde a abordagem em tempo discreto at o clculo da norma euclidiana. Discute-se tambm o efeito de limitaes de hardware no processo de estimaco.

O Captulo 5 apresenta os ensaios estticos realizados e as simulaes feitas no MATLAB para averiguar a eficincia do mtodo proposto. O captulo tambm apresenta algumas limitaes encontradas.



Finalmente, no capítulo 6 são apresentadas as conclusões obtidas neste trabalho. Discute-se também a possibilidade de trabalhos futuros e o que se pode melhorar para tornar o sistema mais eficiente para ser implementado em uma plataforma móvel.

## 2 EMBASAMENTO

### 2.1 Arranjo de sensores

Na proposta deste trabalho, para estimar a direção de um sinal sonoro, será empregado um arranjo de microfones que irá receber o sinal sonoro. Este arranjo é importante porque um único microfone não é capaz de fornecer informações suficientes para estimar a DOA. Quando se analisa o mesmo sinal visto de dois ou mais microfones, o atraso sofrido pelo sinal torna-se uma variável importante, e a estimação deste atraso, tanto em tempo quanto em fase, é o primeiro passo para determinar a direção da fonte sonora.

Neste trabalho, abordou-se o arranjo linear uniforme devido à sua simplicidade. Este consiste em um conjunto de  $N$  elementos, igualmente espaçados a uma distância  $d$  um do outro e dispostos em um segmento de reta AB. A direção de chegada do sinal corresponde ao ângulo de azimute  $\theta$ , formado pela interseção do sinal sonoro  $s(t)$  com a reta perpendicular ao eixo formado pelo segmento de reta. O modelo do arranjo linear pode ser visualizado na Figura 2.1.

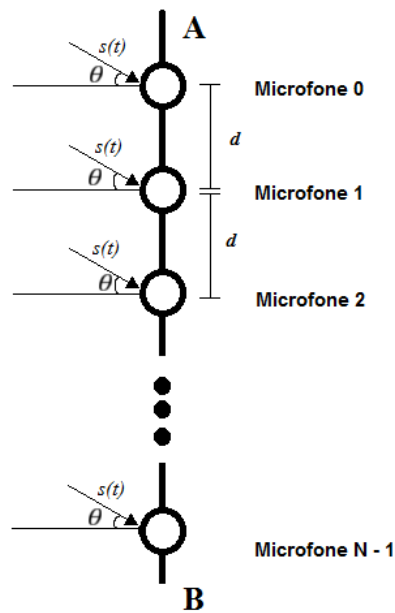


Figura 2.1: Sinal  $s(t)$  incidindo sobre arranjo linear uniforme de  $N$  microfones

Conforme a frente de onda, considerada aqui como uma onda plana, incide sobre o arranjo linear, cada microfone percebe uma diferente versão defasada do sinal. Tal defasagem fica evidente na Figura 2.2, que mostra que o sinal  $s(t)$  deve percorrer uma distância adicional de  $d \cdot \sin(\theta)$  até chegar ao segundo microfone,  $2d \cdot \sin(\theta)$  para chegar ao terceiro, e assim sucessivamente.

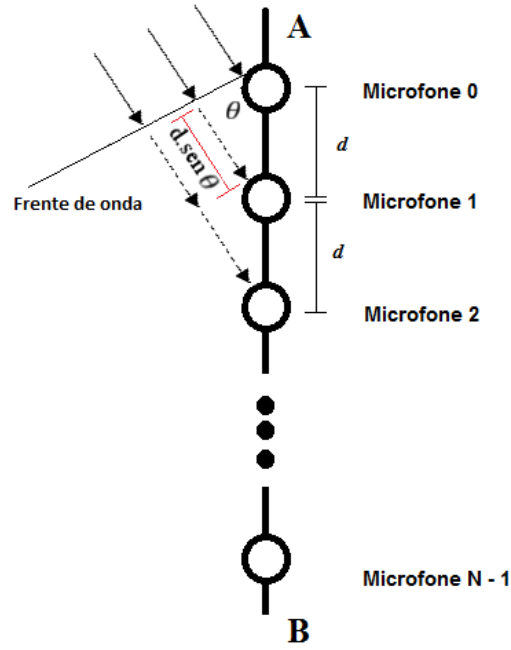


Figura 2.2: Geometria para uma frente de onda plana

Considerando o microfone 0 como referência, o atraso temporal  $\tau$  sofrido pelo sinal que é percebido pelo microfone subsequente depende da velocidade da onda e da distância extra que o sinal deve percorrer, conforme é mostrado na equação (2.1):

$$\tau = \frac{d \cdot \sin \theta}{v} \quad (2.1)$$

Em que  $v$  é a velocidade do som,  $d$  é a distância entre os microfones e  $\theta$  é o azimute. Naturalmente, a defasagem angular pode ser obtida a partir do atraso temporal, como mostra a equação (2.2):

$$\varphi = \omega \cdot \tau = 2\pi f \cdot \frac{d \cdot \sin \theta}{v} \quad (2.2)$$

Finalmente, a velocidade do som  $v$  pode ser expressa como uma relação entre comprimento de onda e frequência, por se tratar da velocidade de uma onda. A equação (2.3) mostra como a defasagem depende também de  $\lambda$ , o que mostra que sinais de frequências distintas sofrerão diferentes atrasos.

$$\varphi = 2\pi \cdot \frac{d \cdot \sin \theta}{\lambda} \quad (2.3)$$

Nota-se que a defasagem mostrada na equação (2.3) é a do microfone 1 em relação ao microfone 0, que aqui se considera como referência. Estendendo a análise, a defasagem do microfone 2 será o dobro da primeira, enquanto que a do microfone 3 será o triplo da primeira.

Assim sendo, pode-se construir uma equação matricial como a equação (2.4), na qual  $x_i(t)$  é a saída do  $i$ -ésimo microfone, em um total de  $M$  sensores, e as defasagens estão representadas pelas exponenciais complexas.

$$\begin{bmatrix} x_0(t) \\ x_1(t) \\ x_2(t) \\ \vdots \\ x_{M-1}(t) \end{bmatrix} = \begin{bmatrix} 1 \\ e^{-j\varphi} \\ e^{-2j\varphi} \\ \vdots \\ e^{-(M-1)j\varphi} \end{bmatrix} s(t) + \begin{bmatrix} n_0(t) \\ n_1(t) \\ n_2(t) \\ \vdots \\ n_{M-1}(t) \end{bmatrix} \quad (2.4)$$

Na equação (2.4) foi adicionado o ruído branco  $n(t)$  ao sinal de entrada que chega em cada microfone, visto que este é intrínseco ao sistema. A equação matricial, por sua vez, é válida desde que  $s(t)$  seja um sinal em banda estreita, podendo ser escrito da forma  $s(t) = b(t)e^{j\omega_0 t}$ , em que  $b(t)$  é limitado em banda. Para o método desenvolvido, isto será de extrema importância, e será explicado na seção 2.3.

A equação (2.4) descreve a situação específica em que  $s(t)$  é um sinal sonoro vindo de uma única fonte. No entanto, para o caso em que há uma quantidade  $N$  de fontes sonoras, a equação matricial generalizada é a equação (2.5). Nota-se que a matriz de defasagem agora é uma matriz  $M \times N$ , englobando defasagens de todos os  $N$  sinais chegando em cada um dos  $M$  microfones.

$$\begin{bmatrix} x_0(t) \\ x_1(t) \\ x_2(t) \\ \vdots \\ x_{M-1}(t) \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ e^{-j\varphi_0} & e^{-j\varphi_1} & & e^{-j\varphi_{N-1}} \\ e^{-2j\varphi_0} & e^{-2j\varphi_1} & & e^{-2j\varphi_{N-1}} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-(M-1)j\varphi_0} & e^{-(M-1)j\varphi_1} & \dots & e^{-(M-1)j\varphi_{N-1}} \end{bmatrix} \cdot \begin{bmatrix} s_0(t) \\ s_1(t) \\ s_2(t) \\ \vdots \\ s_{N-1}(t) \end{bmatrix} + \begin{bmatrix} n_0(t) \\ n_1(t) \\ n_2(t) \\ \vdots \\ n_{M-1}(t) \end{bmatrix} \quad (2.5)$$

O vetor que relaciona os  $N$  sinais emitidos aos  $M$  microfones é denominado *steering vector*, ou vetor volante. Este é dependente da direção  $\theta$  de chegada do sinal, e está descrito na equação (2.6).

$$a(\theta_N) = \begin{bmatrix} 1 \\ e^{-j\varphi_N} \\ e^{-2j\varphi_N} \\ \vdots \\ e^{-(M-1)j\varphi_N} \end{bmatrix} \quad (2.6)$$

## 2.2 Ambiguidade e distância entre sensores

Para evitar que haja ambiguidade na estimação, é crucial que o módulo do ângulo do atraso seja menor ou igual a 180 graus. Sendo assim, ao se aplicar tal condição na equação (2.3) e considerando que o maior valor possível para um seno é 1, obtém-se a equação (2.7).

$$d \leq \frac{\lambda}{2} \quad (2.7)$$

A relação acima é conhecida como a versão espacial do Teorema da Amostragem de *Nyquist*, e fornece a condição de espaçamento entre sensores para que a estimação de DOA seja feita corretamente.

Considerando um sinal sonoro em banda estreita centrado em  $f = 1$  kHz e tendo a velocidade do som  $v = 340$  m/s, obtém-se pela equação (2.7) uma distância entre sensores adjacentes de  $d = 17$  cm. Esta é a distância que oferece a melhor resolução. Entretanto, um sistema embarcado está sujeito a limitações espaciais, como foi o verificado pelo autor. Neste trabalho, considerando as dimensões da plataforma móvel, a distância escolhida para espaçar os microfones utilizados foi de  $d = 15$  cm, a fim de permitir a instalação correta da plataforma.

## 2.3 Bandas de frequência

Como foi afirmado na seção 2.1, a equação (2.3) supõe que, em um arranjo linear de sensores, o sinal de entrada é de banda estreita, ou seja, há uma única frequência central  $f_c$  que concentra a maior parte da energia do sinal. Um sinal em banda larga, por sua vez, possui energia distribuída em uma determinada faixa de frequências, e cada uma delas sofrerá um atraso diferente ao chegar nos diversos microfones. Conforme afirmado anteriormente, o método desenvolvido neste trabalho utiliza um arranjo linear para estimar a DOA, e para tanto deve focar na frequência  $f_c$  de um sinal de banda estreita. Caso contrário, a estimação de DOA torna-se impossível pelo método optado neste trabalho. Há métodos na literatura que lidam com o sinal sonoro em banda larga [2], porém estes requerem elevado poder de processamento.

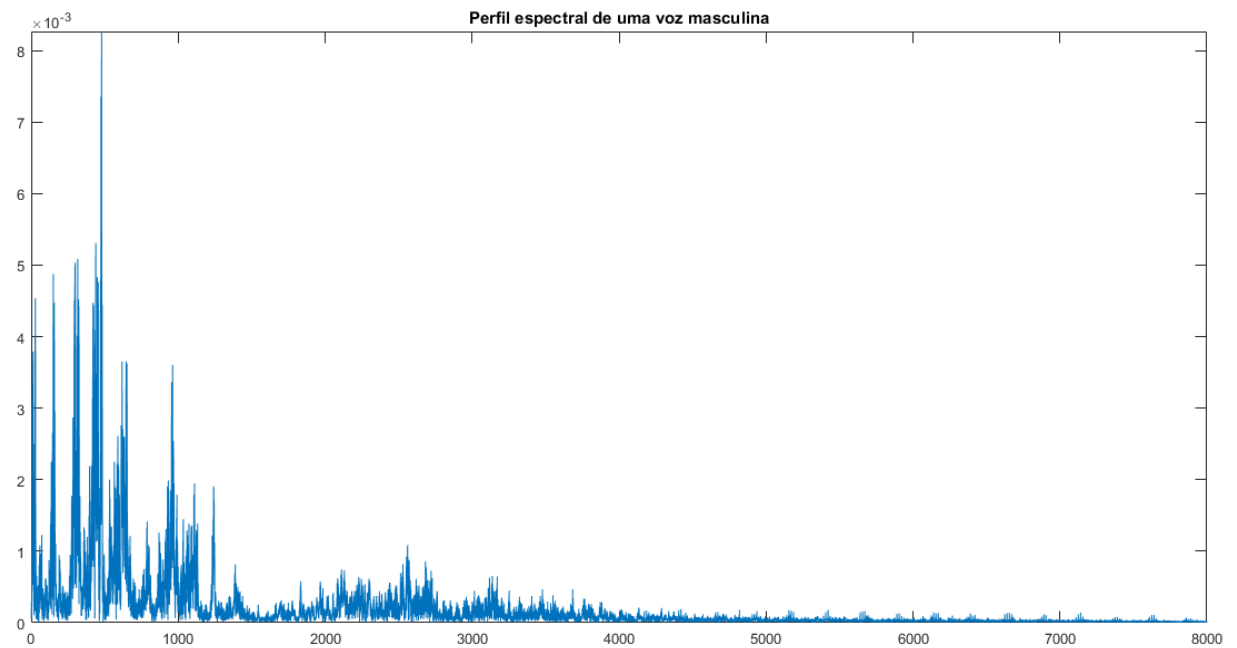


Figura 2.3: Perfil espectral de uma voz masculina

A figura 2.3 mostra o espectro em frequência de um sinal em banda larga  $s(t)$  amostrado a uma taxa de  $f = 16$  kHz e filtrado em  $f_c = 8$  kHz. De maneira simplificada, pode-se afirmar que a banda do sinal de voz humana varia de cerca de 400 Hz até no máximo 4 kHz. Sabe-se que a frequência da voz varia entre indivíduos, o que fica particularmente evidente ao se comparar o espectro de voz de homens e mulheres. Esta característica de banda larga da voz humana dificulta a estimação de DOA, então uma abordagem alternativa faz-se necessária para o método proposto.

Investigou-se que o perfil espectral de assovio, por sua vez, possui variação mínima de pessoa para pessoa. Além disto, como pode ser observado na figura 2.4, possui energia em uma faixa de frequência bastante pequena, próxima de uma frequência principal  $f_d$ . Considerando-se estes fatores, o autor optou por realizar testes com um assovio.

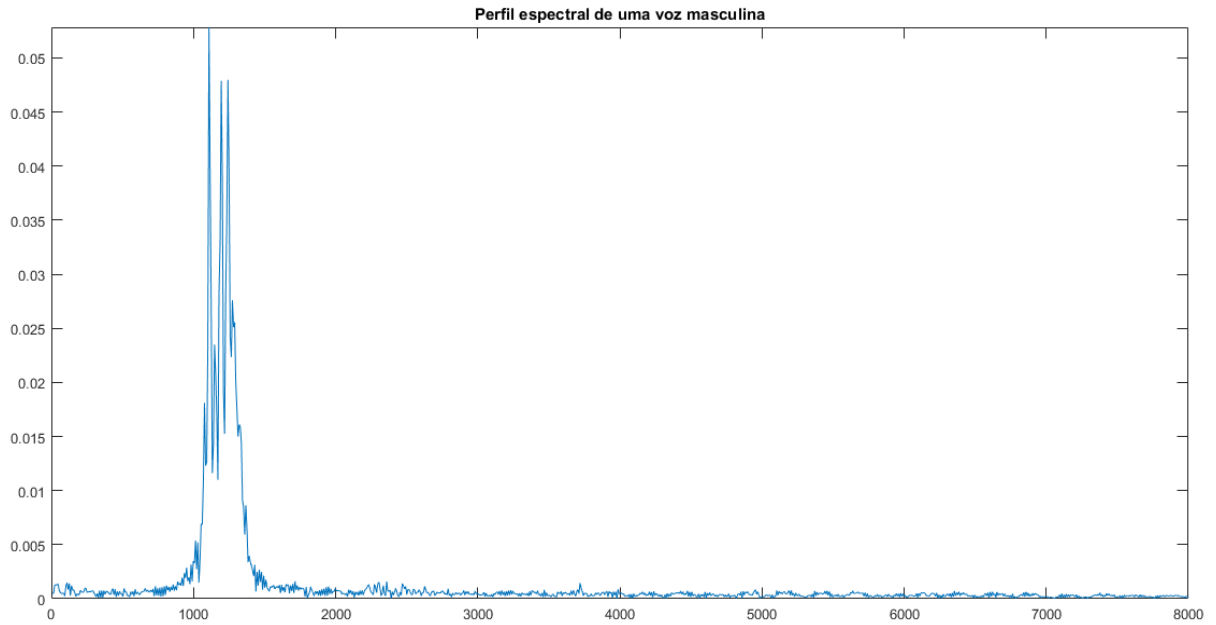


Figura 2.4: Perfil espectral de um assvioio

## 2.4 Filtros digitais

Assim que os sinais analógicos são digitalizados, faz-se necessária a implementação de um filtro digital que selecione a banda de interesse para o processamento digital. É importante observar o perfil espectral típico dos sinais recebidos (no caso, focando no perfil de um assvioio) para projetar um filtro eficiente. Esta seção aborda brevemente os conceitos por trás da implementação de um filtro IIR passa-baixa, focando no contexto deste trabalho em que tal filtro foi aplicado.

Sabe-se que um filtro digital é caracterizado por uma função de transferência descrita no domínio Z, na forma:

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1z^{-1} + b_2z^{-2} + \dots + b_Nz^{-N}}{1 + a_1z^{-1} + a_2z^{-2} + \dots + a_Mz^{-M}}, \quad (2.8)$$

onde  $B(z)$  e  $A(z)$  são os coeficientes que caracterizam a resposta em frequência do filtro. A ordem deste é definida pelo polinômio de maior ordem entre o numerador e o denominador.

O filtro pode ser projetado de maneira intuitiva utilizando-se a ferramenta *filterDesigner* do MATLAB. A ferramenta recebe como parâmetros de entrada:

- O tipo da resposta do filtro (Passa-baixa, passa-faixa, entre outras);
- A ordem do filtro;
- O método de design (IIR, FIR);
- A banda passante do sinal,

Após a definição destes parâmetros, a ferramenta retorna os coeficientes dos polinômios  $A(z)$  e  $B(z)$  da função de transferência  $H(z)$ . Tais parâmetros podem ser aplicados ao sinal no próprio ambiente MATLAB ou então podem ser programados em C no próprio sistema embarcado.

Analisando o perfil espectral apresentado na figura 2.4, construiu-se um filtro IIR – resposta infinita ao impulso, do inglês *Infinite Impulse Response* – Butterworth de 2ª ordem, do tipo passa-baixa e com frequência de corte de 3 kHz, como mostra a figura 2.5. Para fins de visualização em tempo real da resposta, o filtro foi aplicado no ambiente MATLAB.

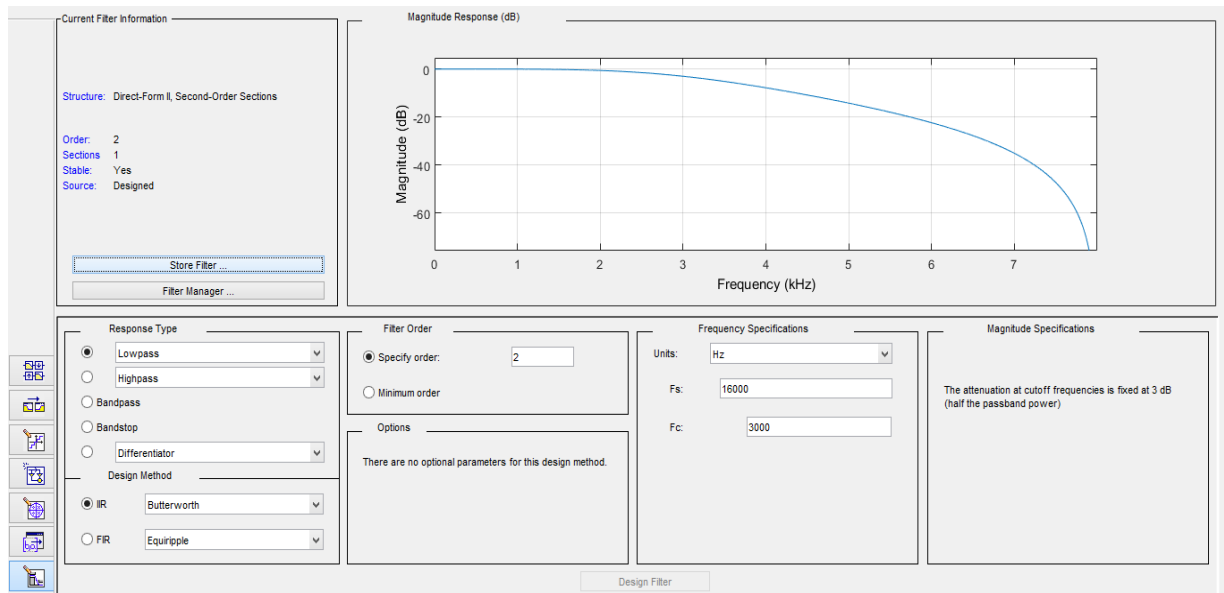


Figura 2.5: Design de filtro IIR em ambiente MATLAB



### 3 HARDWARE

Neste capítulo é apresentada toda a parte de projeto de hardware do sistema, englobando as etapas de aquisição, amplificação, digitalização e processamento do sinal sonoro. O projeto contou com uma série de premissas e objetivos, descritos abaixo:

- Os sensores devem ser idênticos;
- Amplificação linear em módulo e fase com baixa distorção
- Offset DC desprezível;
- Relação Sinal-Ruído (SNR) deve ser a maior possível;

#### 3.1 Microfone e circuito pré-amplificador:

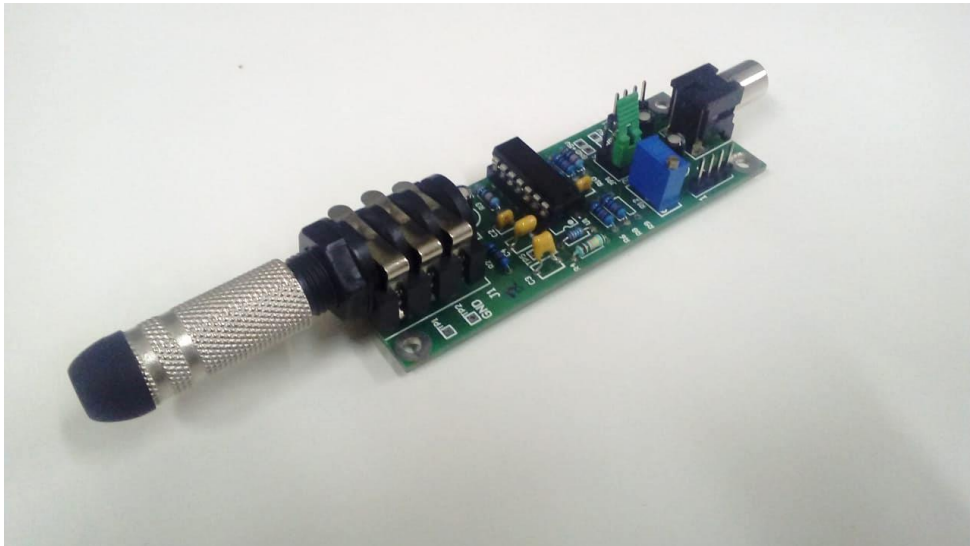


Figura 3.1: Microfone acoplado à placa do circuito utilizado

Neste trabalho, foram utilizados microfones de eletreto omnidimensionais. Estes microfones apresentam SNR de cerca de 62 dB. A curva de resposta em frequência do microfone mostra uma resposta linear para a banda típica de sinais de voz (4kHz), como pode ser observado na figura 3.2.

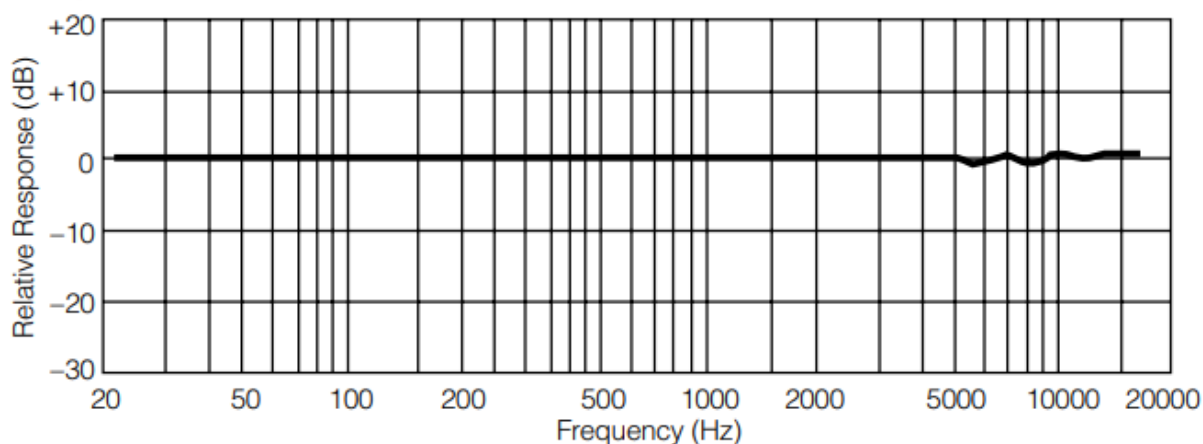


Figura 3.2: Resposta de magnitude versus frequência do microfone de eletreto<sup>1</sup>

O circuito utilizado neste trabalho para captar, condicionar e amplificar o sinal é apresentado na figura 3.3. Este circuito foi montado previamente em uma placa de circuito impresso, e utiliza-se dos amplificadores operacionais contidos no chip LME49740, fabricado pela *National Semiconductor*. Cada chip possui quatro amplificadores.

No circuito, o sinal do microfone é aplicado a um amplificador de instrumentação na sua configuração típica. O ganho é de aproximadamente 270 vezes. Por fim, é aplicado no sinal um filtro passa-baixas na configuração Sallem-key com frequência de corte de 4,378 kHz, e depois um buffer de saída. A saída do buffer é conectada à entrada analógica do Arduino para digitalização do sinal.

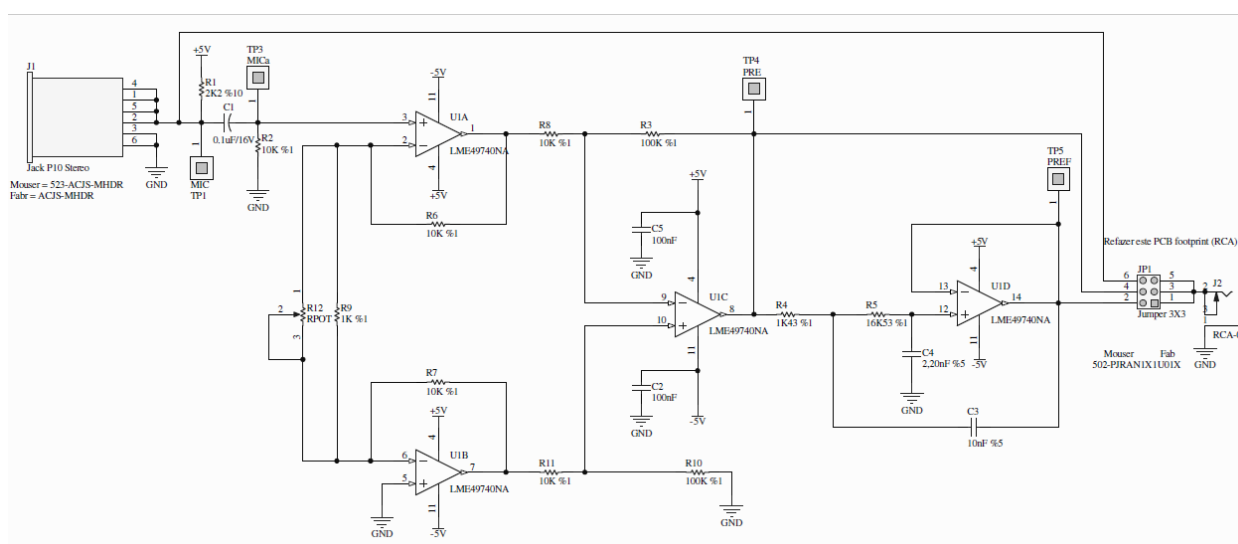


Figura 3.3: Circuito do estágio de pré-amplificação

<sup>1</sup> Curva obtida através do datasheet do eletreto *Panasonic WM-61A*



O circuito é alimentado por uma fonte externa que, neste projeto, consistiu de um conjunto de 4 pilhas de 1,5V, totalizando 6V.

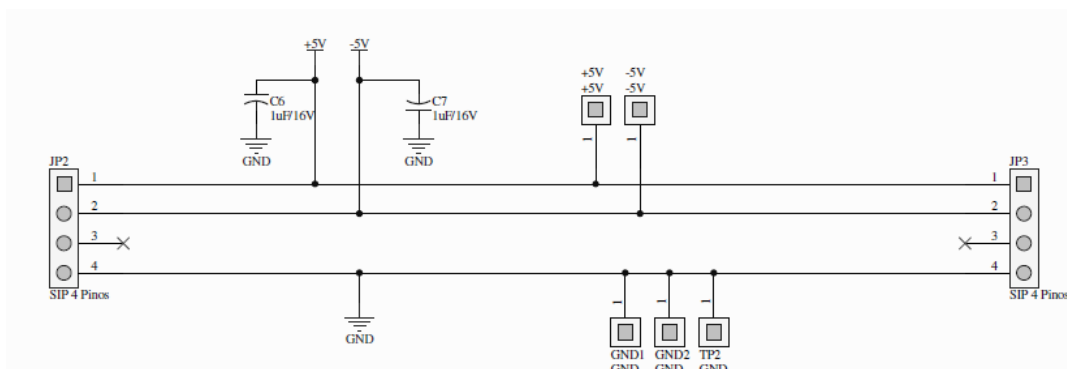


Figura 3.4: Alimentação do circuito

### 3.2 Arduino MEGA 2560

Perseguindo a ideia do desenvolvimento de uma plataforma embarcada acessível, optou-se pela utilização de uma placa Arduino MEGA 2560 para digitalizar e processar o sinal de interesse. A placa, mostrada na figura 3.5, é simples e acessível, e permite o processamento dos dados em tempo real.



Figura 3.5: Placa Arduino MEGA 2560 utilizada no projeto<sup>2</sup>

<sup>2</sup> <https://store.arduino.cc/usa/arduino-mega-2560-rev3>

Os pinos de entrada A0 e A1 são configurados para receber os sinais dos dois microfones que vêm do estágio de amplificação. Como será demonstrado no capítulo 5, quanto maior a taxa de amostragem, maior é a resolução da estimativa do ângulo  $\theta$ . Deseja-se, portanto, obter o maior valor possível para a frequência de amostragem ( $f_{ams}$ ). Contudo, após os primeiros testes verificou-se que, por limitações do Arduino, a digitalização do sinal torna-se ineficiente para taxas maiores que 16 kHz em cada microfone, inserindo elevado nível de ruído no sinal, e em alguns casos até causando erros no processo de digitalização. Como a taxa limite  $f_{ams}$  igual a 16 kHz apresentou resultados bastante satisfatórios, escolheu-se esta taxa para amostrar cada sinal, o que requer a configuração do *timer* do conversor AD em 32 kHz ( $2f_{ams}$ ).

É crucial que os microfones consigam gravar durante tempo suficiente para que se obtenha uma amostra relevante do sinal de interesse (i.e., um assovio). Para tanto, estimou-se, de maneira empírica, que uma janela temporal de 2,5 segundos seria suficiente. O “tempo de gravação”, entretanto, é diretamente limitado pela memória disponível para guardar informação. A uma taxa total de 32 kHz (16 kHz para cada microfone), com uma janela total de tempo de 2,5 segundos, tem-se o número N de amostras a serem obtidas pelo conversor é de:

$$N_{amostras} = 32.000 \frac{amostras}{segundo} \times 2,5 \text{ segundos} = 80.000 \text{ amostras} . \quad (3.1)$$

Quanto ao espaço ocupado em memória, cada amostra digitalizada ocupa 1 Byte, o que significa que a janela proposta requer um espaço disponível de 80 KB. A memória interna do Arduino ATmega2560, entretanto, não é capaz de armazenar toda essa informação, pois possui capacidade de apenas 4 KB. Tendo em vista a taxa de amostragem total utilizada, a memória interna do Arduino permite a gravação em uma janela de tempo de 125 milissegundos, o que é totalmente insatisfatório.

Um aspecto importante da amostragem realizada utilizando apenas uma placa Arduino é que ela não é simultânea. Isto significa que os canais A0 e A1 são amostrados alternadamente a cada batida do relógio do conversor AD, o que resulta em um pequeno atraso entre as amostras. Este atraso é constante, e é um valor conhecido igual ao inverso da frequência do conversor AD, ou seja,  $31,25 \mu s$ . Este valor, uma vez conhecido, pode ser subtraído dos sinais durante a estimativa para eliminar a influência deste atraso.

### 3.3 Memória externa

Para evitar a saturação da memória interna do Arduino, utilizou-se a comunicação serial periférica (SPI) da plataforma para enviar os dados recém-convertidos para uma memória SRAM – memória estática de acesso aleatório, do inglês (*Static Random Access Memory*) externa. A memória utilizada foi a 23LC1024, fabricada pela Microchip. Possui 128 KB de espaço, o que é mais que suficiente para a aplicação deste trabalho. A conexão entre a placa e a memória SRAM é mostrada na figura 3.6, evidenciando os pinos e as funções exercidas segundo o datasheet da memória externa.

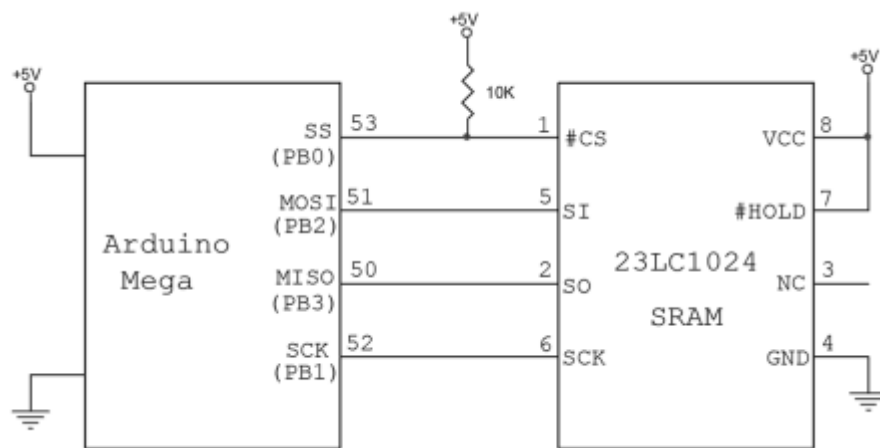


Figura 3.6: Conexão Arduino-SRAM

Deve-se atentar ao tempo que a memória SRAM leva para escrever os dados que vêm do Arduino. Programou-se o SCK da memória, ou *Serial Clock*, para operar em 1 MHz. Isto significa que o processador leva 1  $\mu s$  para escrever 1 bit na SRAM. Evidentemente, 8 bits (1 Byte) levarão 8  $\mu s$  para serem escritos. Vale notar que antes de escrever ou ler na memória, deve-se indicar o endereço em 24-bit da memória onde serão escritos (ou lidos) os dados. Este comando é acessado pelo pino SI durante a subida do relógio. Após a determinação do endereço, pode-se enviar os dados pela porta SPI. No total, o processo de indicar endereço e escrever (bem como indicar o endereço e ler) a memória leva, ao todo, 9  $\mu s$ . Como o ADC leva 31,25  $\mu s$  para digitalizar uma amostra do sinal, a taxa programada no SCK serve perfeitamente.

O processo de digitalização e a escrita na memória SRAM acontecem ao mesmo tempo. Após finalizar ambos, a memória já está disponível para ser lida. O Arduino então acessa a memória, lê os dados e os envia por porta serial para o MATLAB ou para o próprio monitor serial do Arduino, a fim de iniciar o tratamento dos sinais.

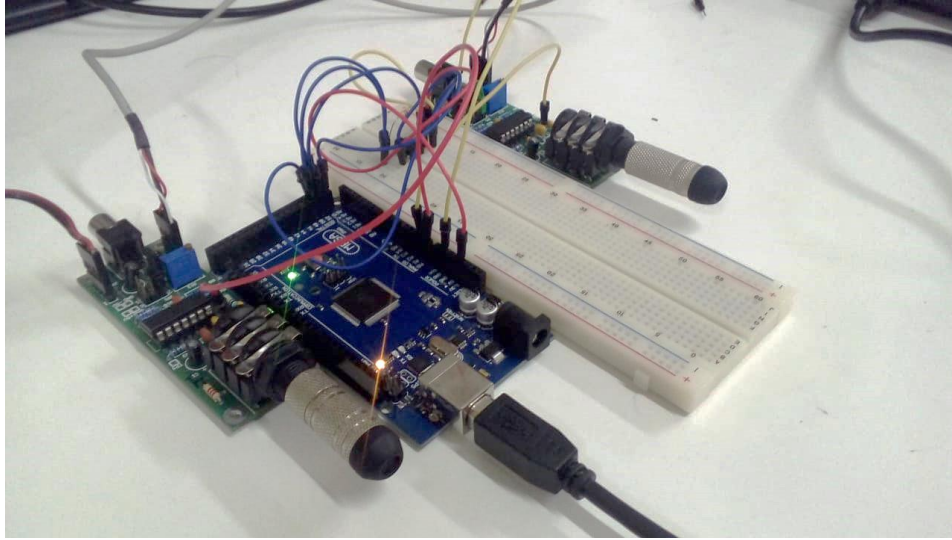


Figura 3.7: Arranjo montado

## 4 MÉTODO DE ESTIMAÇÃO DE DOA

Por simplicidade, o arranjo linear proposto foi composto por apenas dois microfones, dispostos a uma distância de cerca de 15 centímetros um do outro, como mostra na figura 4.1. A fonte sonora, definida com um assvio, encontra-se em uma direção específica, definida pelo azimuth  $\theta$ .

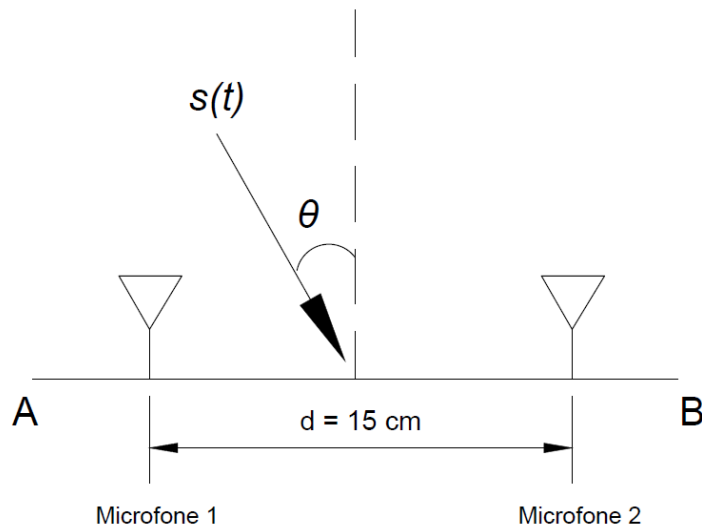


Figura 4.1: Montagem proposta de arranjo com dois microfones

O mesmo assvio é recebido como dois sinais ligeiramente defasados entre si, definidos pelas equações (4.1) e (4.2):

$$m_1(t) = A_1 s(t) + n(t), \quad (4.1)$$

$$m_2(t) = A_2 s(t - \tau) + n(t). \quad (4.2)$$

Nestas, os sinais recebidos pelo primeiro e segundo microfones, respectivamente, são representados por  $m_1(t)$  e  $m_2(t)$ . Além disto,  $n(t)$  representa o ruído branco gaussiano,  $A_1$  e  $A_2$  representam o ganho dos sinais, e  $\tau$  é o atraso temporal do sinal 2 em relação ao primeiro.

#### 4.1 Abordagem em tempo discreto

Os sinais  $m_1(t)$  e  $m_2(t)$ , conforme afirmado em seções anteriores, passam por um processo de conversão analógico-digital, tornando-se sinais em tempo discreto amostrados a uma taxa de  $f_{ams} \left[ \frac{\text{amostras}}{\text{segundo}} \right]$ . Os sinais, a partir deste ponto, já não são mais dependentes do tempo contínuo  $t$ , mas sim de um indexador denominado  $k$ .

Para estimar o azimuth, os sinais em tempo discreto devem ser concentrados em uma janela de  $n$  amostras. Por conveniência, pode-se tratar os sinais como dois vetores discretos,  $\vec{x}_1$  e  $\vec{x}_2$ , como descrito nas equações (4.3) e (4.4):

$$\vec{x}_1 = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad (4.3)$$

$$\vec{x}_2 = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}. \quad (4.4)$$

Em se conhecendo o atraso em tempo discreto entre os dois sinais, é possível calcular o atraso temporal  $\tau$  entre eles. Isto pode ser feito por meio de uma simples análise dimensional, como mostra a equação (4.5):

$$\tau_k = \frac{k}{f_{ams}} = \frac{[amostra]}{\left[ \frac{amostra}{tempo} \right]} = [tempo], \quad (4.5)$$

em que  $k$  representa o atraso relativo entre os dois sinais em tempo discreto. Aplicando o resultado acima na equação (2.1), obtém-se a relação do azimuth  $\theta$  com o atraso  $k$ , descrita conforme a equação (4.6):

$$\tau_k = \frac{k}{f_{ams}} = \frac{d \cdot \sin \theta}{v}. \quad (4.6)$$

Manipulando a equação (4.6), tem-se que:



$$\theta = \text{sen}^{-1}\left(\frac{k \cdot v}{d \cdot f_{ams}}\right), \quad (4.7)$$

em que  $v$  é a velocidade do som (340 m/s),  $d$  é a distância entre os dois microfones (15 cm) e  $f_{ams}$  é a frequência de amostragem do sinal. A partir da equação (4.7), pode-se estimar o azimuth  $\theta$  desde que o valor  $k$  seja corretamente estimado.

Para tanto, os sinais digitalizados devem passar por uma etapa de tratamento, na qual  $\vec{x}_1$  e  $\vec{x}_2$  são enviados do Arduino para um computador por transmissão pela porta serial para passarem por um rápido tratamento, composto por três etapas:

1. Identificação da faixa de maior energia do sinal;
2. Aplicação de um filtro “média-móvel”;
3. Interpolação cúbica;
4. Estimação da direção de chegada.

## 4.2 Primeira etapa: Identificação da faixa de maior energia

O objetivo desta etapa é identificar a faixa de maior relevância do sinal captado pelos microfones, isto é, a faixa que contém a maior energia dos sinais. Uma vez encontrada a faixa de interesse, o restante do sinal é descartado. Desta forma, garante-se que  $\vec{x}_1$  e  $\vec{x}_2$  contenham apenas informação relevante para a estimação. A figura 4.2 mostra uma gravação completa de um sinal sonoro digitalizado pelo Arduino, contendo determinadas faixas com elevada energia, bem como um período considerável de “silêncio”.

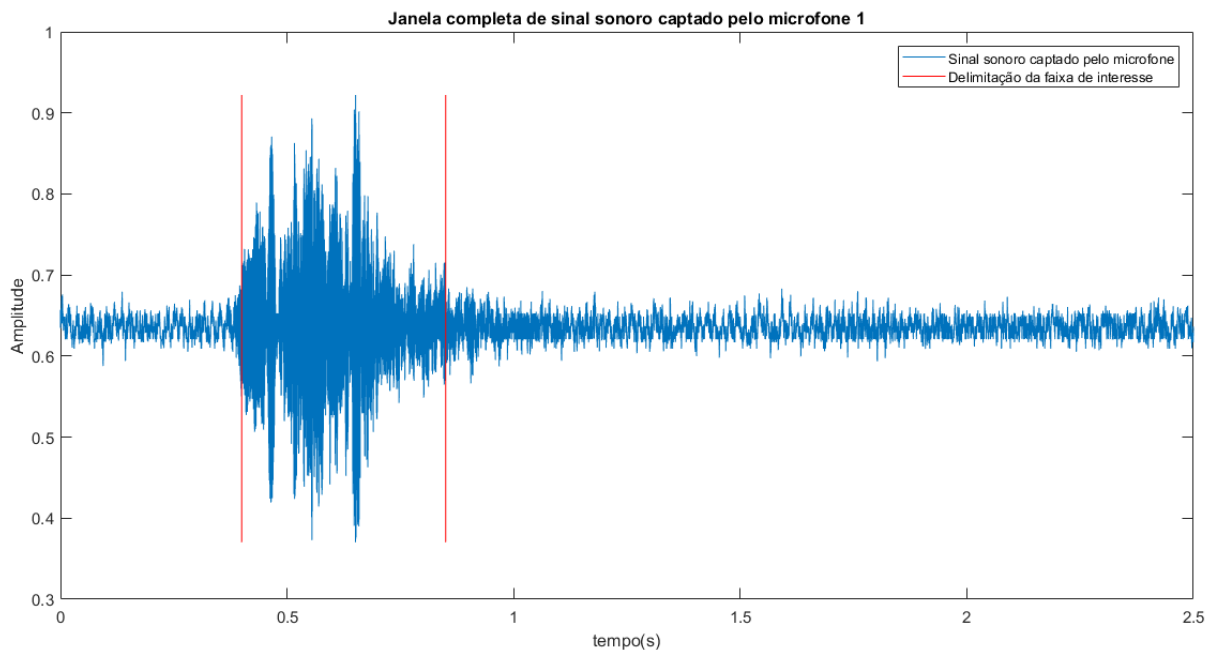


Figura 4.2: Janela completa de sinal sonoro captado por um microfone

Por “informação irrelevante”, entende-se períodos gravados em que nenhum assvio é emitido pela fonte. Como o sinal passa por uma pré-filtragem durante a etapa de amplificação do sinal, a energia contida nos períodos irrelevantes é consideravelmente menor que a da faixa de interesse, o que facilita a abordagem.

Para encontrar a faixa de interesse, deve-se primeiramente tomar como referência um dos vetores que contém o sinal em tempo discreto. Por simplicidade, considerou-se como referência o sinal contido em  $\vec{x}_1$ , que é obtido pelo microfone 1 na figura 4.1. Remove-se a componente DC deste sinal subtraindo-o de sua média. Em seguida, calcula-se o valor absoluto de cada elemento de  $\vec{x}_1$ . Por fim, determina-se um limiar, ou *threshold*, calculado a partir da amostra de maior valor absoluto em  $\vec{x}_1$ . De forma prática, o limiar foi estabelecido com uma fração de 50% deste maior valor. São eliminadas as amostras com valor absoluto menor que este limiar, restando então apenas a faixa de valores dentro do limiar estabelecido.

#### 4.3 Segunda etapa: Aplicação de um filtro de média móvel

Os processos de aquisição e conversão analógico-digital podem inserir ruído considerável no sinal original, e isto pode implicar em erros na determinação da direção de chegada do sinal. Para que isto seja evitado, após a identificação da faixa de energia relevante dos sinais digitalizados, estes passam por um processo de filtragem chamado de “Média Móvel”.

O filtro média móvel de tamanho  $n$  é um filtro FIR – resposta finita ao impulso, do inglês *Finite Impulse Response* – que determina que o valor de determinada amostra  $i$  de um vetor é igual à média entre os últimos valores.

#### 4.4 Terceira etapa: Interpolação por spline cúbica

Como foi citado na seção 3.2, obtenção de uma taxa de amostragem elevada revelou-se uma limitação de hardware, pois a partir de quando se tomava  $f_{ams} > 16 \text{ kHz}$ , o sinal tornava-se excessivamente ruidoso. A taxa de amostragem, entretanto, é um fator crucial na obtenção de uma resolução aceitável na estimação de DOA, conforme será mostrado na seção 4.10.

Para que se alcançasse uma taxa ideal, lançou-se mão de polinômios de interpolação do tipo spline cúbica, que são polinômios de 3º grau e transformam o conjunto de dados de entrada em um novo conjunto, com transições mais suaves entre as amostras. Neste trabalho, a spline cúbica pôde ser facilmente implementada no ambiente MATLAB, retornando como resultado uma versão com uma taxa de amostragem duas vezes maior para o mesmo conjunto de dados.

#### 4.5 Quarta etapa: Estimação da direção de chegada

A medida de similaridade é um número que define quão próximos dois vetores são um do outro. Conforme afirmado anteriormente, este trabalho busca implementar um algoritmo suficientemente rápido e simples, que possa ser implementado em um microprocessador com

capacidade limitada, mas cuja utilização possua um bom custo-benefício. Neste trabalho, investigou-se apenas uma dentre várias formas possíveis de medir tal similaridade. O método escolhido foi o cálculo da norma euclidiana, ou norma  $l^2$ , entre dois sinais  $\vec{x}_1$  e  $\vec{x}_2$ , devido principalmente à sua simplicidade.

#### 4.6 Norma euclidiana

Considerando  $\vec{v}$  como um vetor  $1 \times n$  do tipo:

$$\vec{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \quad (4.8)$$

A norma euclidiana deste vetor é descrita como a raiz quadrada da soma dos valores absolutos elevados ao quadrado de cada elemento do vetor  $\vec{v}$ , conforme demonstra a equação (4.9):

$$|\vec{v}| = \sqrt{\sum_{i=1}^n |v_i|^2}. \quad (4.9)$$

Como deseja-se saber a diferença entre os vetores  $\vec{x}_1$  e  $\vec{x}_2$ , que armazenam os dados digitalizados, define-se a equação (4.10) como:

$$\frac{1}{n} |\vec{x}_1 - \vec{x}_2| = \frac{1}{n} \sqrt{\sum_{i=1}^n |x_{1,i} - x_{2,i}|^2}, \quad (4.10)$$

onde  $x_{1,i}$  e  $x_{2,i}$  representam os  $i$ -ésimos elementos dos vetores que armazenam, respectivamente, os dados do primeiro e segundo microfone. O resultado da equação (4.10) fornece um valor de erro, também chamado de Erro Quadrático Médio, que quantifica a semelhança entre o vetor  $\vec{x}_1$  e o vetor  $\vec{x}_2$ .

#### 4.7 Defasagem em tempo discreto

Como os sinais foram obtidos utilizando dois canais distintos do Arduino, deve-se atentar para o atraso intrínseco do sistema, gerado devido à alternância entre os canais durante a amostragem do sinal. Este atraso, como foi comentado na seção 3.2, é igual a  $31,25 \mu s$ , o que

equivale a exatamente 1 amostra. Uma vez eliminado este atraso, pode-se finalmente executar o algoritmo.

Para obter a defasagem entre os dois sinais em tempo discreto  $\vec{x}_1$  e  $\vec{x}_2$ , deve-se fazer a medida de similaridade entre os dois, aplicando a equação (4.10) em versões deslocadas dos vetores. Para cada deslocamento  $k$  entre os vetores, como mostra a figura 4.3, tem-se um elemento do vetor  $\vec{e}$ , de tamanho  $1 \times n$ , que armazena o cálculo do erro entre os sinais para cada  $k$ .

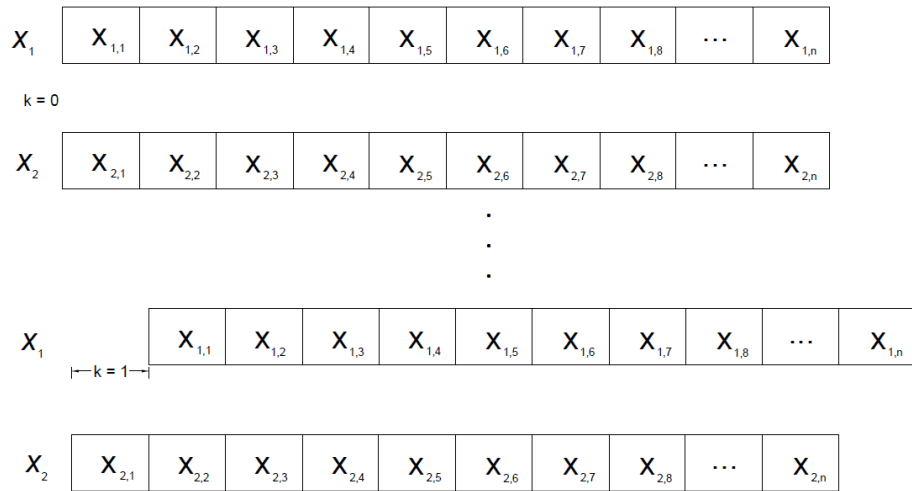


Figura 4.3: Representação do deslocamento em tempo discreto efetuado entre dois sinais de tempo discreto

Analisando a equação (4.10), é evidente que quanto menor o erro, mais próximos são os dois sinais. Sendo assim, ao completar-se o cálculo do erro quadrático médio, busca-se encontrar um valor para o elemento  $k$ , que resulte no menor valor, em módulo, dentre os elementos do vetor  $\vec{e}$ . Este  $k$  representa o atraso em tempo discreto entre os sinais. Este atraso será aplicado na equação (4.7) para a estimação do azimuth.

Como o cálculo da norma euclidiana é realizado em sinais que são aproximadamente periódicos, isto significa que o erro obtido também é periódico, variando entre períodos de grande similaridade e pouca similaridade entre os sinais. Definindo-se as amostras  $k_a$  e  $k_b$  como a primeira e a segunda amostra que contém a maior medida de similaridade (menor erro) em  $\vec{e}$ , é possível calcular a frequência principal de maior energia do assóvio ao se tomar:

$$f = \frac{f_{ams}}{k_a - k_b} \quad (4.11)$$

A figura 4.4 mostra o resultado do cálculo do erro quadrático médio entre dois sinais sonoros digitais  $\vec{x}_1$  e  $\vec{x}_2$ . O cálculo utiliza uma janela de 1.000 amostras, mas para melhor

visualização do ponto de interesse do algoritmo a figura mostra apenas as 300 primeiras amostras. Este ponto encontra-se evidenciado na figura.

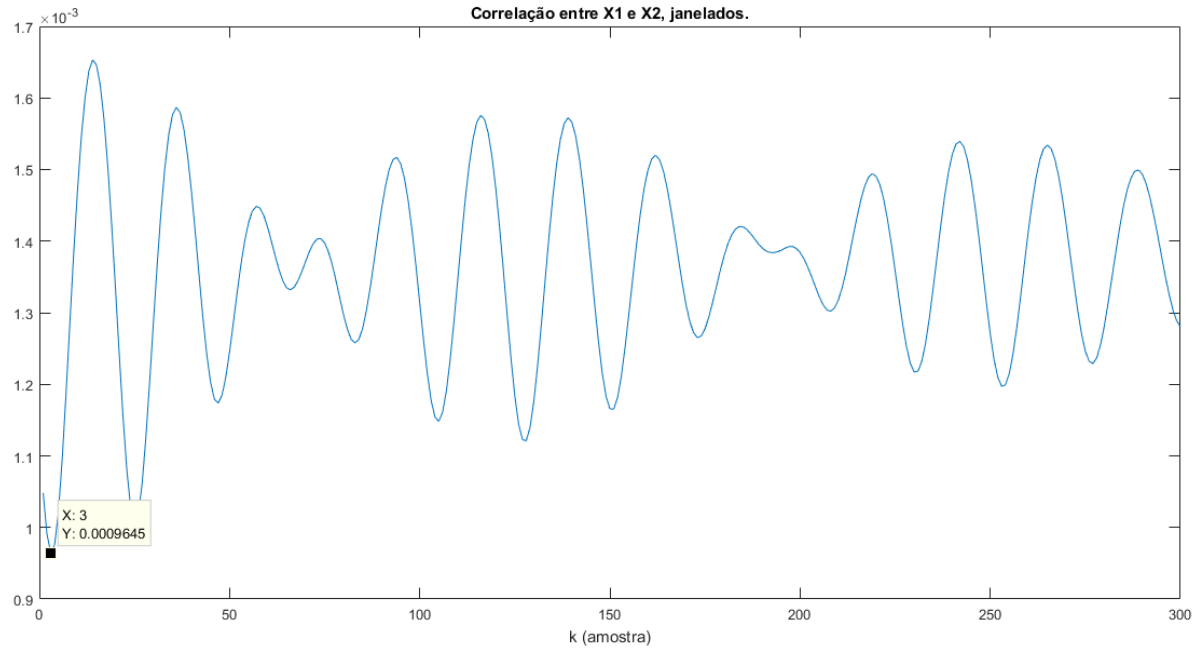


Figura 4.4: Janela de erro quadrático médio calculado entre sinais sonoros captados pelos microfones

#### 4.8 Múltiplas estimativas

Como a abordagem trata de um processo estatístico, é natural que ocorram falhas nas estimativas de  $\theta$ . Por “falha” entende-se resultados grosseiramente diferentes do esperado. Considerando que o objetivo principal deste trabalho é obter uma estimativa válida do azimuth, deve-se evitar estimativas pouco acuradas da melhor forma possível, sem deixar de considerar as limitações de hardware.

Assim, foi proposto que os sinais originais, antes de passarem pelo cálculo do erro entre os dois, sejam limitados a uma janela de 1.000 amostras. Após o cálculo, obtém-se a estimativa do ângulo  $\theta$ , e armazena-se este valor. As últimas amostras utilizadas são descartadas, e seleciona-se o grupo subsequente de 1.000 amostras dos sinais originais filtrados. Repete-se o processo de estimação, e o novo valor de  $\theta$  também é armazenado. Este processo repete-se mais uma vez, e uma média simples é feita entre os três valores de  $\theta$ , conforme a equação (4.12):

$$\theta_{final} = \frac{\theta_1 + \theta_2 + \theta_3}{3} \quad (4.12)$$

#### 4.9 Resolução do estimador e limitações

Da equação (4.7), depreende-se que a estimação de DOA por medidas de similaridade entre os sinais é altamente dependente da taxa de amostragem dos sinais, que por sua vez é uma limitação de hardware.

O primeiro aspecto a ser abordado é a resolução da estimativa, definida pela equação (4.13):

$$\theta_{res} = \text{sen}^{-1} \left( \frac{v}{d \cdot f_{ams}} \right) = \left( \frac{\text{radianos}}{\text{amostra}} \right), \quad (4.13)$$

Como foi discutido nos capítulos anteriores, em se tratando de um sinal analógico de som  $s(t)$ , cada azimuth de incidência  $\theta$  implica em um determinado atraso  $\tau$ , em segundos, entre as versões de  $s(t)$  recebidas por cada microfone. De maneira análoga, em sinais digitais de som, cada azimuth de incidência  $\theta$  resultará em um atraso de  $k$  amostras entre as versões digitalizadas do sinal. Analisando a equação (4.12), percebe-se que a resolução é altamente dependente dos parâmetros  $d$  e  $f_{ams}$ . O espaço disponível para o arranjo de microfones, entretanto, normalmente é um valor fixo, o que faz da distância  $d$  um valor fixo. A resolução da estimativa depende, então, da frequência de amostragem.

Sejam dois sinais sonoros periódicos quaisquer  $s_1(t)$  e  $s_2(t)$ , sendo  $s_2(t)$  uma versão defasada em  $\varphi$  graus em relação ao primeiro. Após serem captados por um arranjo de dois microfones a uma distância  $d = 15\text{cm}$  um do outro, e serem digitalizados a uma taxa de 8 kHz, tornam-se sinais discretos armazenados em dois vetores  $\vec{s}_1$  e  $\vec{s}_2$ . Pela equação (4.12), calcula-se que a resolução do estimador será de  $\theta_{res} = 16,46^\circ$  por amostra. Isto significa que qualquer  $\theta$  de incidência menor que  $16,46^\circ$  será desprezado pelo estimador.

Supondo que os mesmos sinais sejam amostrados a uma taxa de 32 kHz, calcula-se que a resolução do estimador será de  $\theta_{res} = 4,06^\circ$ , que é uma resolução consideravelmente maior que a anterior. Com isto, conclui-se que elevadas frequências de amostragem permitem que o sistema detecte pequenas variações do azimuth  $\theta$ .

Por estar razão optou-se por adequar o hardware para amostrar o sinal na maior taxa possível. Como citado anteriormente, a taxa ótima ficou em torno de 16 kHz. Utilizando-se da interpolação por spline cúbica, tornou-se possível elevar a frequência  $f_{ams}$  para 32 kHz, o que resulta em uma resolução  $\theta = 4,06^\circ$ . Uma variação desta magnitude é considerada pequena, dada a limitação de processamento do Arduino.

A taxa de amostragem também influencia diretamente o segundo aspecto a ser abordado, que é o valor limite de  $k$ . Sabe-se, da trigonometria, que o seno de um ângulo deve ser menor ou igual a 1. Aplicando esta informação na equação (4.6), obtém-se a equação (4.12):

$$k \leq \frac{d \cdot f_{ams}}{v}. \quad (4.12)$$

A equação acima mostra que, a uma distância  $d$  fixa entre os microfones, o atraso  $k$  máximo que pode ser detectado pelo sistema é diretamente proporcional à taxa de amostragem. Na situação proposta por este trabalho, colocando-se os microfones a uma distância de 15 cm, e amostrando o sinal a 32 kHz, o valor limite de atraso é de  $k = 14$ , que resulta em um azimuth máximo de  $\theta = 82,60^\circ$ . Por trigonometria, percebe-se que caso o sistema identifique um atraso maior que 14 amostras, são obtidos, a rigor, ângulos complexos. Estes casos, porém, são ignorados pelo estimador proposto, devendo-se repetir a aquisição.

Após detalhar todas as etapas do método proposto neste trabalho para estimar a direção de chegada de um sinal sonoro, pôde-se construir o fluxograma presente na figura 4.3, que resume todo o processo. Cada bloco foi abordado nos capítulos anteriores, desde a etapa de aquisição até a estimação do ângulo azimuthal. Os próximos capítulos discutirão os resultados observados, e extrairão as conclusões finais do projeto.



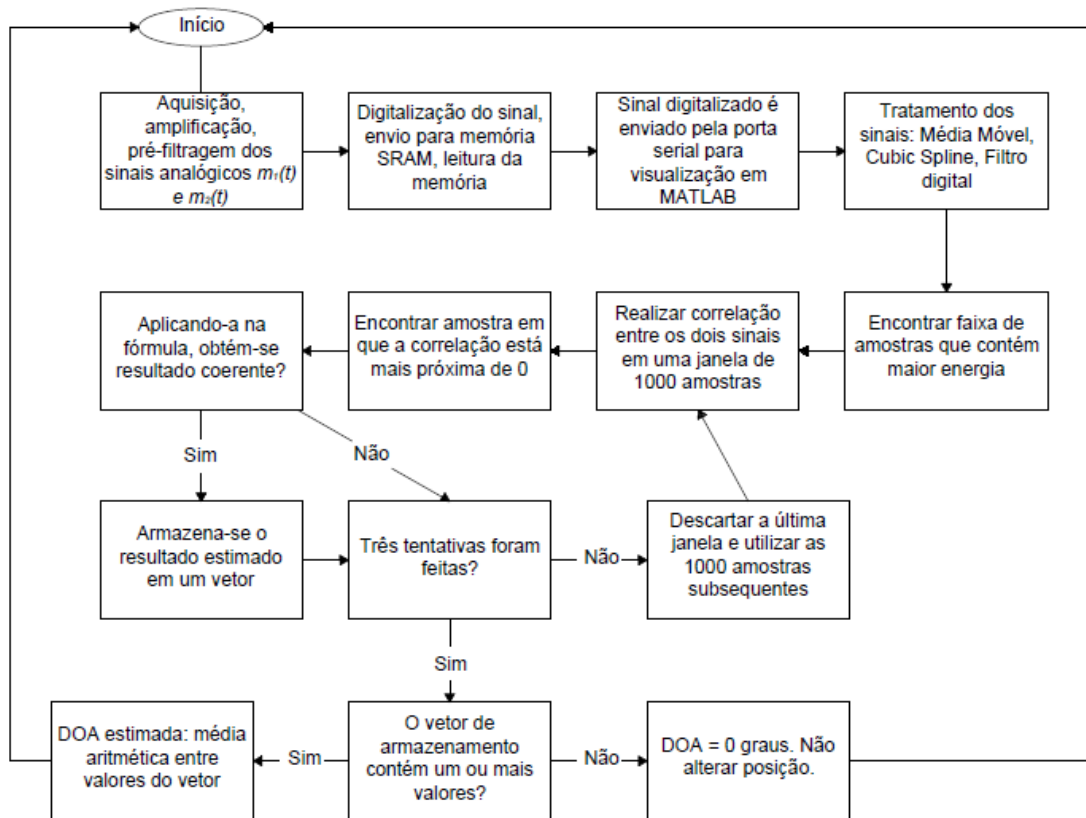


Figura 4.5: Fluxograma que resume o método de estimação de DOA

## 5 RESULTADOS

Este capítulo apresenta os resultados experimentais da estimação de direção de chegada de um sinal sonoro utilizando cálculo de erro quadrático médio. Diversas simulações foram feitas com o software MATLAB para averiguar a eficiência do método proposto.

Foi utilizado um arranjo linear de dois microfones, espaçados a uma distância  $d = 15 \text{ cm}$ . A frequência de amostragem  $f_{ams}$  em cada microfone foi igual a 16 kHz. Os testes foram realizados em ambiente fechado e com baixo ruído. Durante os ensaios, a fonte sonora posicionou-se a  $0^\circ$ ,  $20^\circ$ ,  $50^\circ$  do centro do arranjo. Também se verificou a eficiência do algoritmo em detectar ângulos negativos, como  $-30^\circ$ .

A figura 5.1 mostra o efeito da etapa de tratamento em um dos sinais adquiridos pelo microfone para um azimuth de  $20^\circ$ . Os sinais X1 e X2 correspondem, respectivamente, aos sinais descritos nas equações (4.3) e (4.4). Nota-se, nas imagens, uma remoção perceptível do ruído de conversão, bem como o considerável período de silêncio em que não há nenhum sinal

relevante para o sistema. A figura 5.2 mostra a eficiência do algoritmo que detecta a janela de tempo de interesse no arquivo de som, que está marcada por duas barras verticais vermelhas.

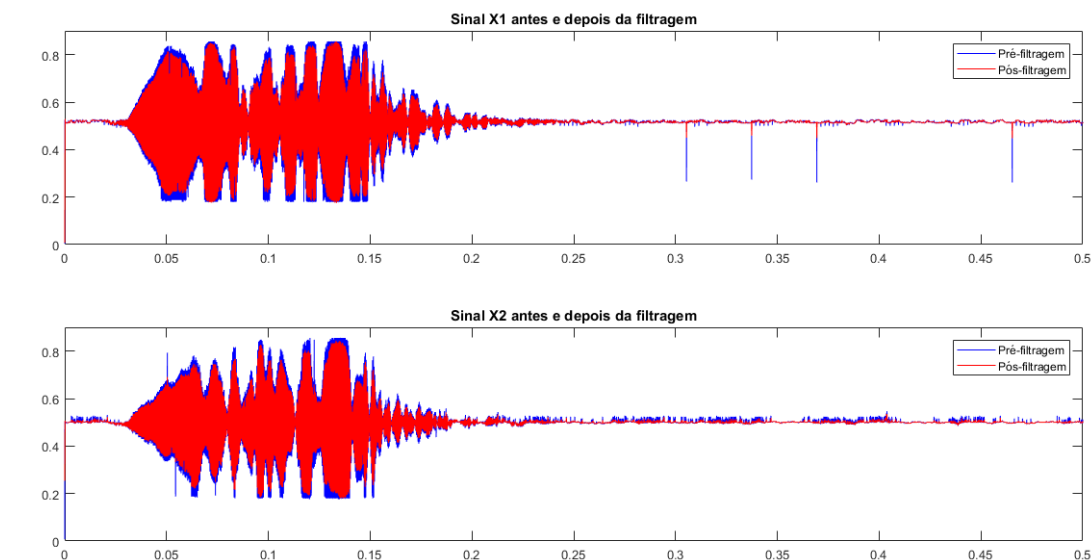


Figura 5.1: Sinais X1 e X2 antes e depois da filtragem

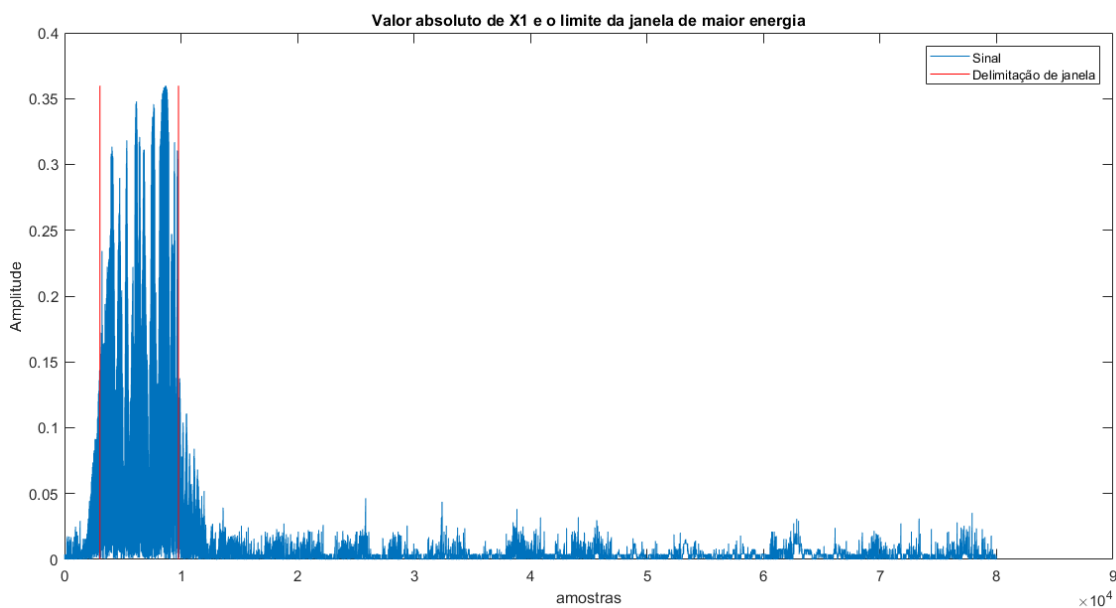


Figura 5.2: Demonstração da detecção da janela de maior energia

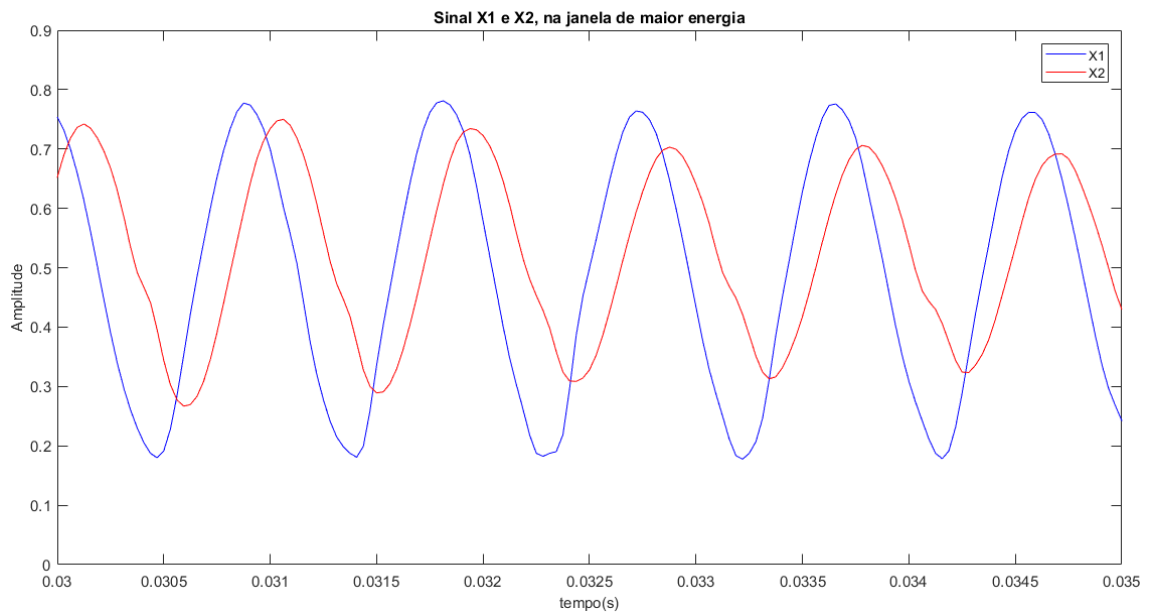


Figura 5.3: Forma de onda ao longo do tempo em que observa a defasagem entre os sinais X1 e X2 em uma aquisição

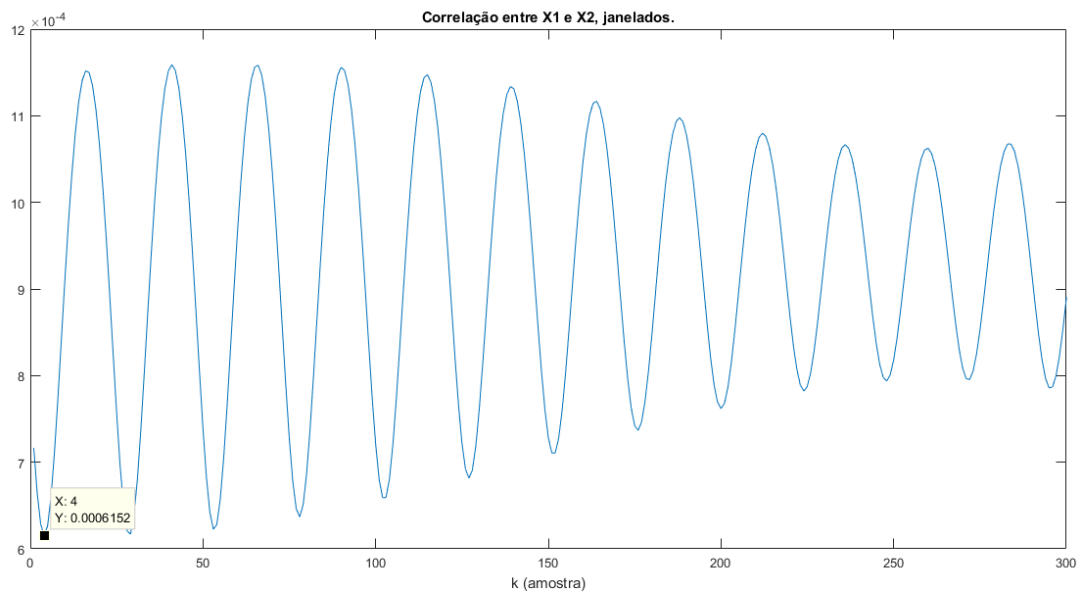


Figura 5.4: Erro quadrático médio entre X1 e X2, calculado em uma simulação de um azimute de  $20^\circ$

A figura 5.5 foi feita utilizando o software MATLAB, com o intuito de vislumbrar graficamente o resultado da estimação do azimute. Posteriormente, pretendia-se utilizar este modelo de representação para indicar em tempo real a detecção da direção de chegada do sinal

sonoro. Isto, porém, não foi possível, devido a problemas verificados na comunicação serial entre Arduino e MATLAB.

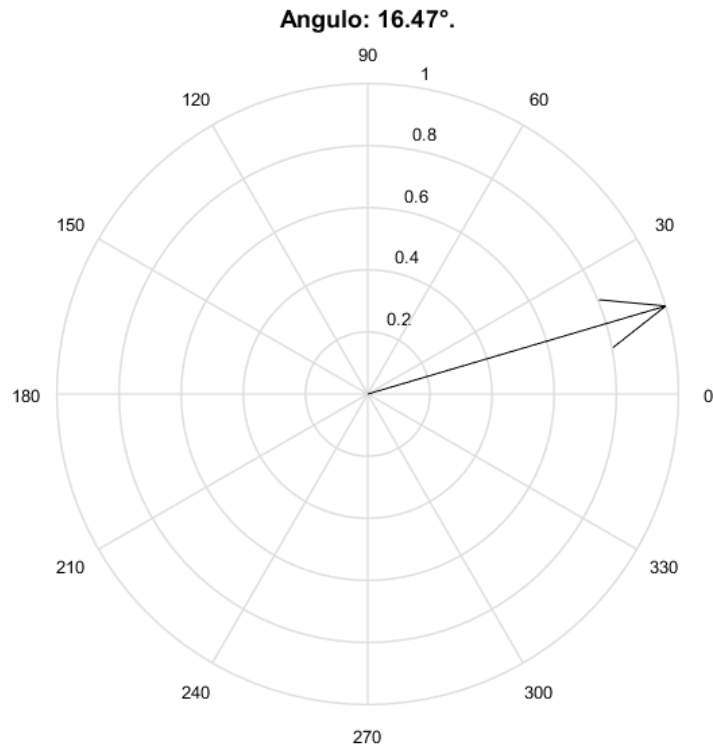


Figura 5.5: Azimute detectado para um ângulo de cerca de 20°

Para validar o algoritmo utilizado, é necessário fazer determinado número de repetições, observando os resultados para que depois possam ser feitas conclusões sobre a eficiência do método. Para tanto, foram escolhidos os seguintes ângulos de referência: 0°, 20°, 50° e -30°. Como demonstrado no diagrama da figura 4.5, cada ensaio obtém dois sinais a partir dos microfones. São feitas três estimativas de azimute a partir de janelas distintas de 1000 amostras dos sinais obtidos. Conforme explicitado na seção 4.9, para a frequência de amostragem e distância entre microfones, o atraso máximo que pode ser identificado pelo sistema é de  $k=14$  amostras. Caso alguma estimativa identifique atraso maior que 14, o azimute resultante não é contabilizado pelo estimador. Finalmente, o resultado do ensaio é a média das estimativas contabilizadas.

Considerando a possibilidade de o estimador não contabilizar nenhuma estimativa válida, o sistema retorna um azimute 0° – em uma aplicação prática, isto significa que o sistema não deve alterar sua posição, pois nenhuma estimativa confiável foi feita.

Por meio da abordagem citada anteriormente, foram realizados 15 ensaios para cada ângulos de referência, analisando empiricamente as estimações de DOA.

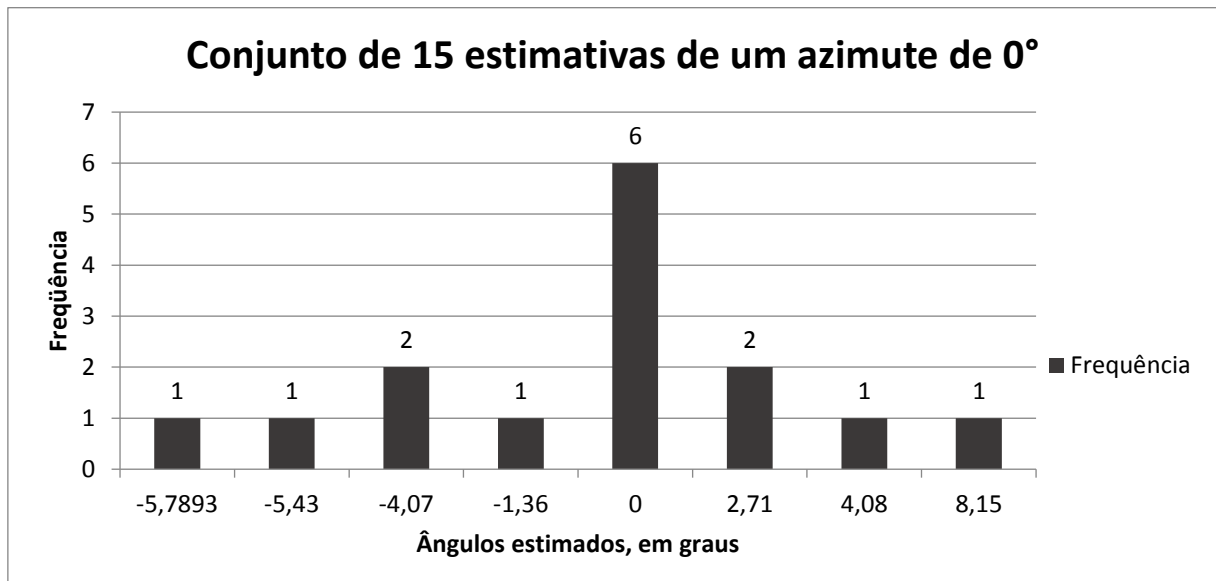


Figura 5.6: Histograma de estimativas para sinais com  $\theta = 0^\circ$

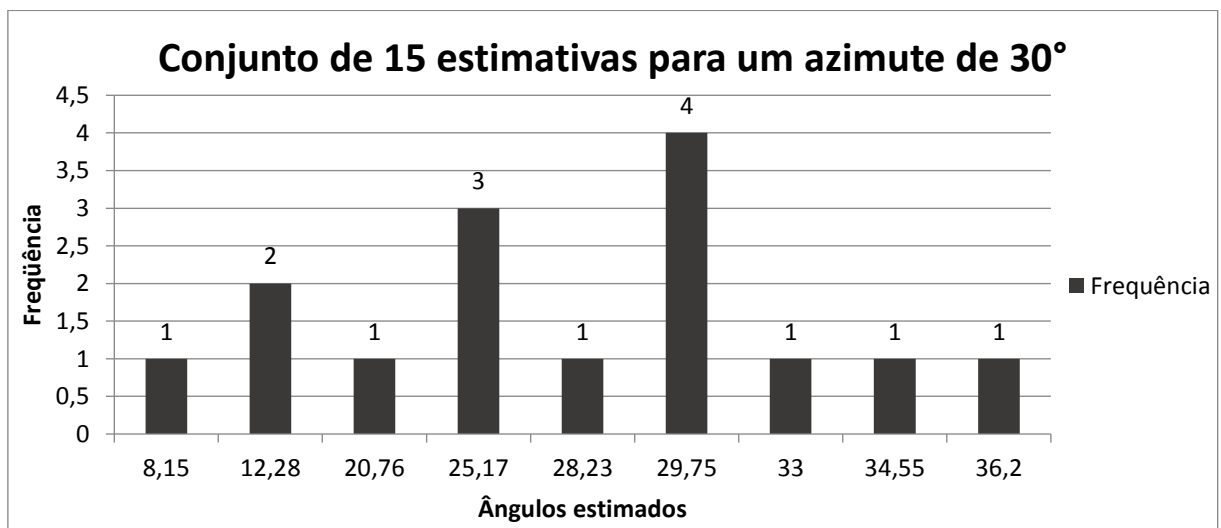


Figura 5.7: Histograma de estimativas para sinais com  $\theta = 30^\circ$

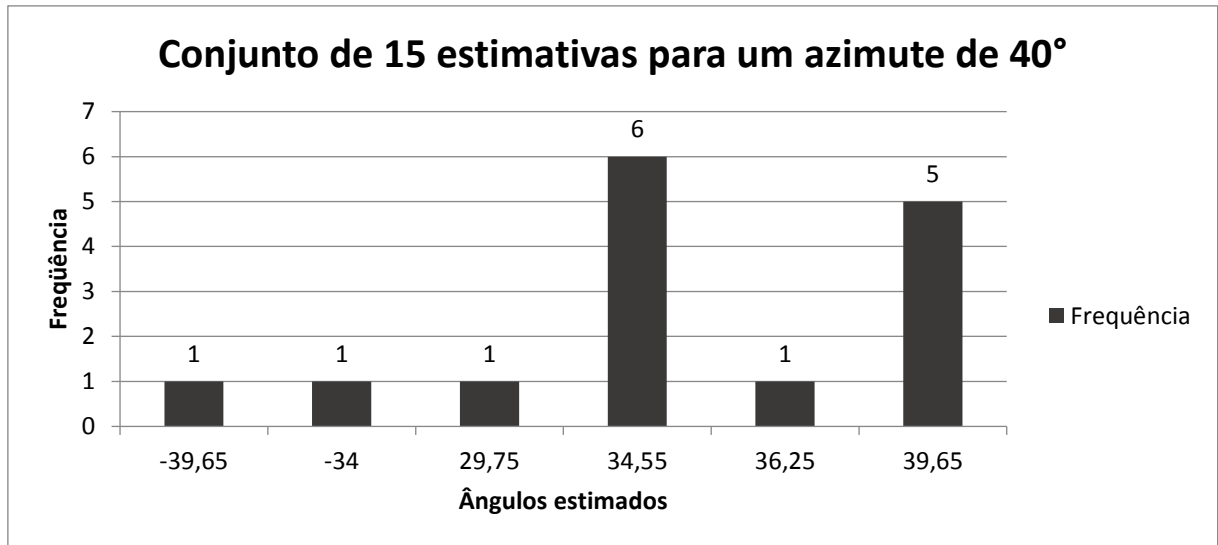


Figura 5.8: Histograma de estimativas para sinais com  $\theta = 40^\circ$

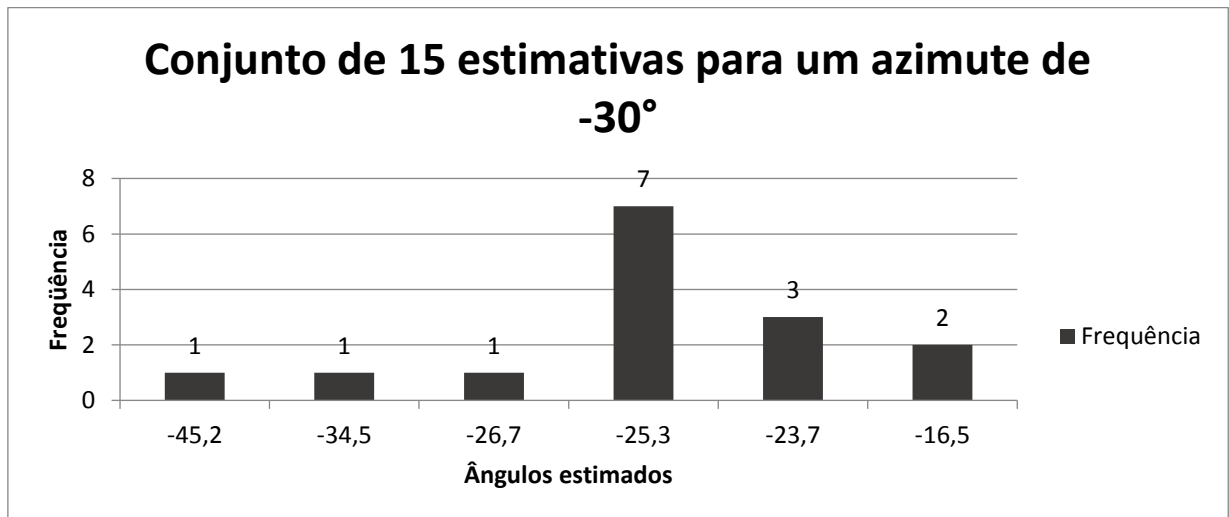


Figura 5.9: Histograma de estimativas para sinais com  $\theta = -30^\circ$

É importante notar como a estimação também funciona para ângulos negativos. Tomando por referência a figura 4.1, temos  $\theta = 0$  quando a fonte do som se encontra alinhada à reta normal ao segmento AB. Então, convencionou-se que  $\theta > 0$  para sinais oriundos de fontes à esquerda da reta normal, enquanto que se tem  $\theta < 0$  quando a fonte se encontra à direita da mesma reta.

## 5.1 Avaliação de resultados

As tabelas 1 e 2 apresentam um resumo dos resultados das aquisições realizadas neste trabalho. Para cada grupo de aquisições, foi feita uma média ponderada pela frequência de ocorrência dos valores estimados pelo algoritmo. Apresenta-se, também, o atraso temporal relativo medido entre os dois sinais.

Tabela 1: Médias das 15 aquisições de DOA para cada ângulo de referência

Referência da fonte sonora	Ângulo azimutal médio estimado	Atraso relativo (ms)
0°	-0,20°	0,0016
30°	25,33°	0,1887
40°	26,53°	0,1970
-30°	-25,84°	0,1923

Tabela 2: Frequência principal do assvio detectada em diferentes situações

Referência da fonte sonora	Frequência principal do sinal (kHz)
0°	1,54
30°	1,65
40°	1,73
-30°	1,79

Como afirmado anteriormente, para cada ângulo de referência foram realizados 15 ensaios. Pelo método de medidas de similaridade, cada ensaio realizou três estimativas de  $\theta$  em diferentes posições do sinal. O ensaio é concluído ao se calcular a média das estimativas utilizadas, e então dá-se início a um novo ensaio. Este procedimento é repetido até que todos os ensaios sejam realizados.

Em cada conjunto de ensaios, também foi estimada a frequência média do sinal  $s(t)$  recebido. Nota-se que a frequência é um valor relativamente constante, possuindo um desvio padrão de apenas 106 Hertz entre as estimativas, com uma média total de 1,68 kHz.

Analisando-se as figuras 5.6 a 5.9, além das tabelas 1 e 2, pode-se notar que o algoritmo funciona e consegue acertar razoavelmente bem os ângulos de chegada. A frequência de boas estimativas é satisfatória para o algoritmo utilizado. Observa-se, porém, que certas estimativas podem se desviar bastante do resultado esperado.

Um caso a se analisar é o das estimativas feitas para um DOA de 40°. Nelas, observa-se uma frequência relativamente alta de resultados próximos ao esperado. Não obstante, os

resultados discrepantes são pontos muito fora da curva, e podem ter consequências variadas, a depender da aplicação. A título de exemplo, um sinal de controle que ordene a um robô que altere sua direção em  $-39^\circ$  pode ter um efeito radicalmente oposto ao sinal que ordena que o mesmo robô gire a  $39^\circ$ .

Para evitar respostas errôneas do sistema, é necessário implementar um algoritmo que seja capaz de ignorar estimativas demasiadamente discrepantes. Analisando os gráficos obtidos, percebe-se que estimativas muito distantes do valor desejado, apesar de possíveis, não são muito frequentes em relação às outras. Sendo assim, é interessante que o sistema consiga fazer várias aquisições e identificar e armazenar conjuntos de estimativas com uma diferença de não mais que, por exemplo,  $10^\circ$  umas das outras. O sistema então assume que o conjunto com o maior número de elementos é a resposta desejada, e retorna como resposta a média entre os elementos de tal conjunto.

Esta solução possui duas limitações, ambas interligadas: a aplicação desejada e o poder de processamento disponível. Esta última é crucial para obter uma resposta precisa em um curto espaço de tempo, visto que não é desejável que o estimador demore muito a obter um valor de azimuth. Embora não seja o caso abordado neste trabalho, uma fonte sonora que se movimenta no espaço, por exemplo, poderá fazer com que o estimador identifique ângulos muito diferentes uns dos outros, se o sistema não conseguir amostrar o sinal e estimar o azimuth em uma velocidade suficientemente alta.

## **5.2 Limitações do projeto**

Uma das principais limitações do projeto está na incapacidade de o arranjo detectar o sentido do sinal. Este problema possui diversas causas, uma delas sendo a utilização de sensores



omnidirecionais no arranjo. Estes detectam sinais sonoros vindos de todas as direções, o que dificulta a detecção do sentido do sinal.

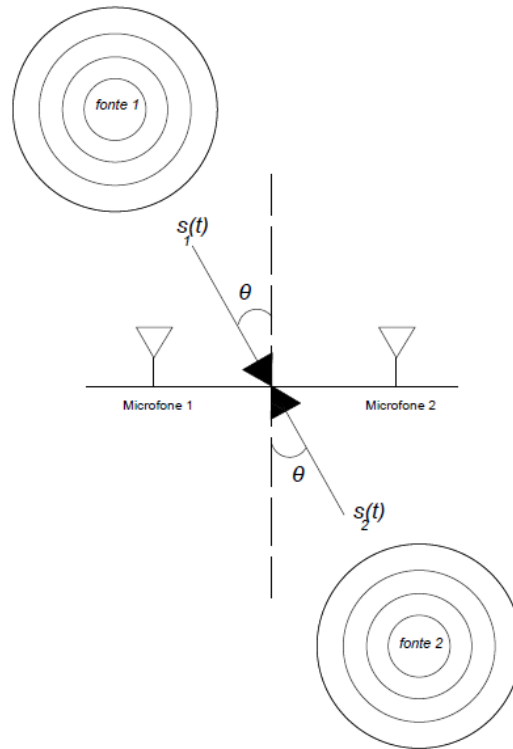


Figura 5.10: Demonstração de como o mesmo ângulo pode valer para duas fontes diferentes

A segunda causa é o próprio algoritmo, pois embora o azimuth  $\theta$  possa ser detectado, o algoritmo não consegue extrair nenhuma informação que permita saber se a fonte de som se encontra à frente do arranjo ou atrás dele. A fim de verificar a existência deste problema, foram realizados dois breves ensaios que buscam avaliar a estimação de DOA pelo arranjo quando a fonte se encontra atrás dele. Considerando a figura 5.1 como referência, o azimuth da fonte sonora neste ensaio seria, aproximadamente, igual a  $120^\circ$ . O ângulo  $\theta$  estimado, entretanto, foi  $\theta = -31,35^\circ$ , como mostra a figura 5.11.

Um segundo problema enfrentado no desenvolvimento deste projeto foi a comunicação MATLAB – Arduino. Desejava-se construir uma interface gráfica simples que permitisse a visualização das estimativas de  $\theta$  em tempo real. Contudo, o autor teve vários problemas na comunicação serial entre a placa e o software, obtendo respostas muito lentas por parte do MATLAB, o que limitou a utilização deste. A capacidade de processamento do Arduino também é um fator limitante, pois muitos dos algoritmos podem levar tempo demasiado para retornarem estimativas válidas.

## 6 CONCLUSÃO

Este trabalho focou no desenvolvimento de algoritmos rápidos, implantados em um processador acessível, que permitam estimar a direção de chegada de um sinal sonoro. Embora este trabalho tenha sido feito visando a estimação de DOA em controle de robôs por som, as aplicações dos métodos aqui descritos vão muito além disto, dentre elas: próteses auditivas inteligentes, rastreamento de disparos de armas, sonares, antenas diretivas, entre outras.

Ao longo do período de desenvolvimento do trabalho, verificou-se a eficiência de uma diversidade de componentes na montagem do arranjo. Chegou-se a desenvolver um pequeno circuito de pré-amplificação, porém este sofria com baixa relação sinal-ruído, o que prejudicava muito o estágio de digitalização do sinal. O novo circuito, fornecido pelo orientador do projeto, foi um trabalho desenvolvido por alunos da Universidade de Brasília.

Como foi discutido no capítulo anterior, estimativas muito discrepantes de  $\theta$  podem gerar problemas, a depender da aplicação. Por isto, propõe-se a utilização de um processador mais veloz, que consiga fazer repetidos ensaios em curto espaço de tempo, além de medidas estatísticas com os resultados.

O aumento na velocidade de processamento também acarreta no aumento da taxa de amostragem do sinal. Uma taxa elevada poderia evitar que se utilizasse os polinômios spline em um sinal com elevado número de amostras. Além disto, pode também propiciar o desenvolvimento de arranjos menores, mais dinâmicos, que permitam estimar a direção do sinal e localização relativa da fonte deste.

Para trabalho futuros, deseja-se testar a funcionalidade do algoritmo em ambiente dinâmico, em que se deseja rastrear uma fonte sonora que se move. Isto será bastante útil, por exemplo, na implementação em uma plataforma móvel seguidora de assovios. Para tanto, será necessário averiguar a praticidade da implementação do algoritmo aqui desenvolvido diretamente no microprocessador, o que também pode exigir maior poder de processamento.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Carvalho, G. A., “Desenvolvimento de plataforma embarcada para estimação da direção de chegada utilizando medidas de similaridade”. Projeto de Iniciação Científica, UnB, Brasília, 2014
- [2] Gontijo, A.T., “Estimação da Direção de Chegada em Tempo Real por Arranjo de Microfones”. Tese de Mestrado, UnB, Brasília, ago. 2010
- [3] LV, X., ZHANG, M., YUAN, G., et al., “Robot sound source search strategy based on multi-blackboard model”. In: Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 633–638, Tianjin, China, December 2010
- [4] N. Dezhi, C. Changxing, et al, “Study on New DOA Estimation Method based-on ULA Under Gaussian Noise”. 3rd International Conference on Multimedia Technology (ICMT 2013), Air-Force Engineering University, Science College, Xi’ an , Shaanxi, 710051, China
- [5] P. Stoica, Z. Wang, and J. Li, “Robust capon beamforming,” IEEE Signal Processing Letters, vol. 10, no. 6, June 2003.
- [6] S. N. Bhuiya, F. Islam, and M. A. Matin, “Analysis of Direction of Arrival Techniques Using Uniform Linear Array”, International Journal of Computer Theory and Engineering, Vol. 4, No. 6, December 2012
- [7] Scola, C.F., Ortega, M. D. B., “Direction of arrival estimation – A two microphones approach”, Blekinge Institute of Technology, September 2010
- [8] Z. Chen, G. Gokeda, and Y. Yu, Introduction to Direction-of-Arrival Estimation, Artech House, vol. 34, no. 37, pp. 46-47, 2010.